Attorney Docket No. **09792909-6437** (formerly 09798423-0006) PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Patent Application of:	)	Group Art Unit: **2161**
	)	
Akira Nonaka, et al.	)	Examiner: **Frantz Coby**
	)	
Application No. **09/856,276**	)	Confirmation No.: **5130**
	)	
Filed: **October 2, 2001**	)	
	)	
For: DATA PROVIDING SYSTEM AND METHOD THEREFOR	)	
	)	

MAIL STOP APPEAL BRIEF - PATENTS
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPELLANTS' AMENDED APPEAL BRIEF

Dear Sir:

Appellants respectfully submit this Amended Appeal Brief in response to the Notification of Non-Compliant Appeal Brief mailed July 13, 2006.

I. REAL PARTY IN INTEREST

The real party in interest in the present appeal is the Assignee, Sony Corporation, a Japanese Corporation. The Assignment was recorded in the U.S. Patent and Trademark Office at Reel 012223, Frame 0442.

II. RELATED APPEALS AND INTERFERENCES

There are no related appeals and no related interferences.

III. STATUS OF CLAIMS

Claims 1-5, 71, 140, 141, 287 and 288 are pending in this application. Claims 6-70, 72-139 and 142-286 have been cancelled. The present Appeal is directed to claims 1-4, 71, 140, and 287-288 that were rejected under 35 U.S.C. § 102(b) as being anticipated by Linehan et al. (U.S. Patent No. 5,495,533), and claims 5 and 141 that were rejected under 35 U.S.C. § 103(a) as being unpatentable over Linehan et al. (U.S. Patent No. 5,495,533) in view of Kravitz et al. (U.S. Patent No. 6,738,905) in a final office action dated September 23, 2005.

IV. STATUS OF AMENDMENTS

There are no pending amendments. However, appellants reserve the right to submit an amendment to correct noted typographical errors that do not affect the appeal.

V. SUMMARY OF CLAIMED SUBJECT MATTER

Claim 1 is directed to a data providing system for distributing content data from a data providing apparatus to a data processing apparatus and managing the data providing apparatus and the data processing apparatus by a management apparatus. (See page 75, line 1 through page 79, line 20.) The management apparatus prepares a key file storing encrypted content key data and encrypted usage control policy data indicating a content of rights such as usage permission conditions of the content data. (See page 87, lines 8-14 and page 75, lines 13-19.) The data providing apparatus provides the content data encrypted by using the content key data. (See page 108, lines 5-11.) The data processing apparatus decrypts the content key data and the usage control policy data stored in the key file and determines the handling of the content data

based on the decrypted usage control policy data. (See page 123, line 18 through page 127, line 14.)

Claim 71 is directed to a data providing method for distributing content data from a data providing apparatus to a data processing apparatus and managing the data providing apparatus and the data processing apparatus by a management apparatus. (See page 75, line 1 through page 79, line 20.) The management apparatus prepares a key file storing encrypted content key data and encrypted usage control policy data indicating a content of rights such as usage permission conditions of the content data. (See page 87, lines 8-14 and page 75, lines 13-19.) The data providing apparatus provides the content data encrypted by using the content key data. (See page 108, lines 5-11.) The data processing apparatus decrypts the content key data and the usage control policy data stored in the key file and determines the handling of the content data based on the related decrypted usage control policy data. (See page 123, line 18 through page 127, line 14.)

Claim 140 is directed to a data providing apparatus which is managed by a management apparatus and distributes content data to a data processing apparatus. (See page 75, line 1 through page 79, line 20.) The data providing apparatus receives a key file storing encrypted content key data and encrypted usage control policy data indicating a content of rights such as usage permission conditions of said content data and distributes a module storing a content file storing the content data encrypted by using said content key data and said key file received from said management apparatus to said data processing apparatus. (See page 107, line 15 through page 108, line 11.)

Claim 141 is directed to a data processing apparatus managed by a management apparatus. (See page 75, line 1 through page 79, line 20.) The data processing apparatus utilizes content data and receives a module containing a key file storing encrypted content key data and encrypted usage control policy data indicating a content of rights such as usage permission conditions of said content data and a content file storing the content data encrypted by using said content key data. (See page 107, line 15 through page 108, line 11.) The data processing apparatus also determines at least one between a purchase form and an usage form of said content data based on said usage control policy data, and transmitting a log data indicating the log of the determined at least one of the related purchase form and usage form to said management apparatus. (See page 127, line 12 through page 130, line 13.)

Claim 287 is directed to a data providing method which is managed by a management apparatus and distributes content data to a data processing apparatus. (See page 75, line 1 through page 79, line 20.) The method includes receiving a key file storing encrypted content key data and encrypted usage control policy data indicating a content of rights such as usage permission conditions of said content data, and distributing a module storing a content file storing the content data encrypted by using said content key data and said key file received from said management apparatus to said data processing apparatus. (See page 107, line 15 through page 108, line 11.)

Claim 288 is directed to a data processing method managed by a management apparatus. (See page 75, line 1 through page 79, line 20.) The method includes utilizing content data and receiving a module containing a key file storing encrypted content key data and encrypted usage control policy data indicating a content of rights such as usage permission conditions of said

content data and a content file storing the content data encrypted by using said content key data. (See page 107, line 15 through page 108, line 11.) The method also includes determining at least one between a purchase form and an usage form of said content data based on said usage control policy data, and transmitting a log data indicating the log of the determined at least one of the related purchase form and usage form to said management apparatus. (See page 127, line 12 through page 130, line 13.)

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

1. Claims 1-4, 71, 140, and 287-288 stand rejected under 35 U.S.C. § 102(b) as anticipated by Linehan et al. (U.S. Patent No. 5,495,533).

2. Claims 5 and 141 stand rejected under 35 U.S.C. § 103(a) as obvious over Linehan et al. in view of Kravitz et al. (U.S. Patent No. 6,738,905).

VII. ARGUMENT

Claims 1-4, 71, 140, and 287-288 are patentable over Linehan et al., and claims 5 and 141 are patentable over Linehan et al. in view of Kravitz et al.

A. The Claimed Invention

Claim 1 is directed to a data providing system for distributing content data from a data providing apparatus to a data processing apparatus and managing the data providing apparatus and the data processing apparatus by a management apparatus. The management apparatus prepares a key file storing encrypted content key data and encrypted usage control policy data indicating a content of rights such as usage permission conditions of the content data. The data

providing apparatus provides the content data encrypted by using the content key data. The data processing apparatus decrypts the content key data and the usage control policy data stored in the key file and determines the handling of the content data based on the decrypted usage control policy data.

Claims 2-5 depend from claim 1.

Claim 71 is directed to a data providing method for distributing content data from a data providing apparatus to a data processing apparatus and managing the data providing apparatus and the data processing apparatus by a management apparatus. The management apparatus prepares a key file storing encrypted content key data and encrypted usage control policy data indicating a content of rights such as usage permission conditions of the content data. The data providing apparatus provides the content data encrypted by using the content key data. The data processing apparatus decrypts the content key data and the usage control policy data stored in the key file and determines the handling of the content data based on the related decrypted usage control policy data.

Claim 140 is directed to a data providing apparatus which is managed by a management apparatus and distributes content data to a data processing apparatus receiving a key file storing encrypted content key data and encrypted usage control policy data indicating a content of rights such as usage permission conditions of said content data, and distributing a module storing a content file storing the content data encrypted by using said content key data and said key file received from said management apparatus to said data processing apparatus.

Claim 141 is directed to a data processing apparatus managed by a management apparatus and utilizing content data, receiving a module containing a key file storing encrypted

content key data and encrypted usage control policy data indicating a content of rights such as usage permission conditions of said content data and a content file storing the content data encrypted by using said content key data, determining at least one between a purchase form and an usage form of said content data based on said usage control policy data, and transmitting a log data indicating the log of the determined at least one of the related purchase form and usage form to said management apparatus.

Claim 287 is directed to a data providing method which is managed by a management apparatus and distributes content data to a data processing apparatus, receiving a key file storing encrypted content key data and encrypted usage control policy data indicating a content of rights such as usage permission conditions of said content data, and distributing a module storing a content file storing the content data encrypted by using said content key data and said key file received from said management apparatus to said data processing apparatus.

Claim 288 is directed to a data processing method managed by a management apparatus and utilizing content data, receiving a module containing a key file storing encrypted content key data and encrypted usage control policy data indicating a content of rights such as usage permission conditions of said content data and a content file storing the content data encrypted by using said content key data, determining at least one between a purchase form and an usage form of said content data based on said usage control policy data, and transmitting a log data indicating the log of the determined at least one of the related purchase form and usage form to said management apparatus.

B. Claims 1-4, 71, 140 And 287-288 Are Patentable

In the Final Office Action, claims 1-4, 71, 140, and 287-288 were rejected under 35 U.S.C. § 102(b) as being anticipated by Linehan et al. (U.S. Patent No. 5,495,533), and claims 5 and 141 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Linehan et al. (U.S. Patent No. 5,495,533) in view of Kravitz et al. (U.S. Patent No. 6,738,905). The Examiner has not made an adequate showing to support his rejections.

Linehan et al. is directed to a security system that automatically manages keys used for encryption or message authentication of data files or individual entries in databases. (See col. 6, lines 8-10.) The personal key archive security system uses two components: the Personal Key Client component on a user computer and the Personal Key Server. (See col. 6, lines 17-20.) The Personal Key Server maintains a Personal Key Database that contains certain information required to decrypt files or check their message authentication. (See col. 7, lines 6-9.)

In Linehan et al., each data file is encrypted by the Personal Key Client at the time the file is created. (See col. 7, lines 30-33.) The Personal Key Server Database contains an entry for each file that is encrypted, and each entry contains the key used to encrypt the corresponding file, the name of the owner of the file, and the access control list containing the names of any of the users who are permitted to access the file. (See col. 7, lines 39-45.) In Linehan et al., the access control list is not encrypted. Thus, Linehan et al. neither discloses nor suggests that the management apparatus prepares a key file storing encrypted usage control policy data indicating a content of rights such as usage permission conditions of the content data, or that the data processing apparatus decrypts the usage control policy data stored in the key file and determines

the handling of the content data based on the decrypted usage control policy data, as required by the claims.

In the Advisory Action, the Examiner stated that

> Linehan provides an automated management system for managing keys to encrypt and decrypt stored data on the computing system. In addition, Linehan provides a mechanism showing the organization of file headers used with key archives that primarily incorporates an access control list (See Linehan et al. Figure 8). The access control list primarily indicates content of rights including usage permission conditions of content data because the access control list includes list of users permitted to access the file (See Linehan et al. Col. 8, lines 57-65).

The Examiner does not appear to dispute that the usage control policy data in Linehan et al. is not encrypted before it is stored in the key file.

Moreover, in Linehan et al., when a file is created, the Personal Key Client sends the ticket of the creator, along with the file's name and creation date, to the Personal Key Server. (See col. 7, lines 47-49.) The Personal Key Server generates a file encryption key, creates a new entry in the database, and responds to the Personal Key Client with the file encryption key. (See col. 7, lines 49-52.) The Personal Key Client then uses the key to encrypt the data as it is written to the file. (See col. 7, lines 52-53.)

When a file is accessed, the Personal Key Client sends the ticket of the accessor, the file's name, and the file's creation date to the Personal Key Server. (See col. 7, lines 54-57.) The Personal Key Server retrieves the appropriate entry in the database and checks the identity of the accessor as provided in the ticket against the file owner's name and access control list in the database entry. (See col. 7, lines 57-60.) If the accessor is either the owner or one of the users named in the access control list, the Server sends the file encryption key back to the Personal Key Client. (See col. 7, lines 60-63.) The Personal Key Client uses the key to decrypt the data

as it is read from the file. (See col. 7, lines 63-64.) Thus, in Linehan et al., the Personal Key Server that generates the file encryption key and maintains the Personal Key Server Database does not store the encrypted file. Accordingly, Linehan et al. does not disclose or suggest a management apparatus that prepares a key file storing encrypted content key data and encrypted usage control policy data indicating a content of rights such as usage permission conditions of said content data, as required by the claims. The Examiner does not appear to dispute this fact.

In view of the foregoing, Appellants respectfully submit that claims 1-4, 71, 140, 287 and 288 are patentable and the application is in condition for allowance.

C. Claims 5 and 141 Are Patentable

As discussed above, Linehan et al. does not disclose or suggest various limitations that are required by the claims. Thus, it would not have been obvious to one skilled in the art at the time of the invention to modify Linehan et al. with the disclosure of Kravitz et al. to derive claim 5 or claim 141, both of which include these limitations.

In view of the foregoing, Appellants respectfully submit that claims 5 and 141 are patentable and the application is in condition for allowance.

D. Conclusion

Appellants respectfully submit that the subject matter of the claims on appeal is not disclosed or suggested by Linehan et al. Thus, the Examiner has not made an adequate showing of anticipation with respect to the subject matter of the rejected claims. Appellants, therefore, respectfully request reversal of the Examiner's decision to reject claims 1-4. 71, 140 and 287-288 under 35 U.S.C. § 102(b) over Linehan et al., and claims 14 and 25 under 35 U.S.C. § 103(a) as

being unpatentable over Linehan et al. in view of Misra, and respectfully request allowance of all pending claims.

Respectfully submitted,

Dated: August 11, 2006

By: _____

Marina N. Saito
Registration No. 42,121
SONNENSCHEIN NATH & ROSENTHAL LLP
P.O. Box 061080
Wacker Drive Station, Sears Tower
Chicago, Illinois 60606-1080
(312) 876-8000

VIII. CLAIMS APPENDIX

1. (Previously Presented) A data providing system for distributing content data from a data providing apparatus to a data processing apparatus and managing said data providing apparatus and said data processing apparatus by a management apparatus, wherein

said management apparatus prepares a key file storing encrypted content key data and encrypted usage control policy data indicating a content of rights such as usage permission conditions of said content data,

said data providing apparatus provides said content data encrypted by using said content key data, and

said data processing apparatus decrypts said content key data and said usage control policy data stored in said key file and determines the handling of said content data based on the decrypted usage control policy data.

2. (Previously Presented) A data providing system as set forth in claim 1, wherein said management apparatus adds signature data for verification to the key file.

3. (Original) A data providing system as set forth in claim 1, wherein said data providing apparatus prepares a content file storing the content data and provides the content file to the data processing apparatus.

4. (Previously Presented) A data providing system as set forth in claim 3, wherein said data providing apparatus adds signature data for verification to the content file.

5. (Original) A data providing system as set forth in claim 1, wherein

the data providing apparatus prepares usage control policy data and sends it to said management apparatus,

said data processing apparatus determines at least one of the purchase form and the usage form of the distributed content data based on the usage control policy data and sends log data showing the log of at least one of the purchase form and the usage form decided to said management apparatus, and

said management apparatus performs profit distribution processing for distributing the profit obtained along with the purchase and usage of the content data in the data processing apparatus to the interested parties of the data providing apparatus based on the received log data.

6-70. (Cancelled).

71. (Previously Presented) A data providing method for distributing content data from a data providing apparatus to a data processing apparatus and managing said data providing apparatus and said data processing apparatus by a management apparatus, wherein

said management apparatus prepares a key file storing encrypted content key data and encrypted usage control policy data indicating a content of rights such as usage permission conditions of said content data,

said data providing apparatus provides said content data encrypted by using said content key data, and

said data processing apparatus decrypts said content key data and said usage control policy data stored in said key file and determines the handling of said content data based on the related decrypted usage control policy data.

72-139. (Cancelled).

140. (Previously Presented) A data providing apparatus which is managed by a management apparatus and distributes content data to a data processing apparatus,

receiving a key file storing encrypted content key data and encrypted usage control policy data indicating a content of rights such as usage permission conditions of said content data, and

distributing a module storing a content file storing the content data encrypted by using said content key data and said key file received from said management apparatus to said data processing apparatus.

141. (Previously Presented) A data processing apparatus managed by a management apparatus and utilizing content data,

receiving a module containing a key file storing encrypted content key data and encrypted usage control policy data indicating a content of rights such as usage permission conditions of said content data and a content file storing the content data encrypted by using said content key data,

determining at least one between a purchase form and an usage form of said content data based on said usage control policy data, and

transmitting a log data indicating the log of the determined at least one of the related purchase form and usage form to said management apparatus.

142-286. (Cancelled).

287. (Previously Presented) A data providing method which is managed by a management apparatus and distributes content data to a data processing apparatus,

receiving a key file storing encrypted content key data and encrypted usage control policy data indicating a content of rights such as usage permission conditions of said content data, and

distributing a module storing a content file storing the content data encrypted by using said content key data and said key file received from said management apparatus to said data processing apparatus.

288. (Previously Presented) A data processing method managed by a management apparatus and utilizing content data,

receiving a module containing a key file storing encrypted content key data and encrypted usage control policy data indicating a content of rights such as usage permission conditions of said content data and a content file storing the content data encrypted by using said content key data,

determining at least one between a purchase form and an usage form of said content data based on said usage control policy data, and

transmitting a log data indicating the log of the determined at least one of the related purchase form and usage form to said management apparatus.

X. EVIDENCE APPENDIX

Appellants attach hereto copies of the patents to (1) Linehan et al. (U.S. Patent No. 5,495,533), and (2) Kravitz et al. (U.S. Patent No. 6,738,905), which were relied upon by the Examiner in his rejection entered on February 16, 2005.

US005495533A

United States Patent [19]

Linehan et al.

[11] Patent Number: 5,495,533

[45] Date of Patent: Feb. 27, 1996

[54] **PERSONAL KEY ARCHIVE**

[75] Inventors: **Mark H. Linehan**, Yorktown Heights, N.Y.; **Nicholas J. Simicich**, Boca Raton, Fla.; **Gene Y. Tsudik**, Thalwil, Switzerland

[73] Assignee: **International Business Machines Corporation**, Armonk, N.Y.

[21] Appl. No.: **235,578**

[22] Filed: **Apr. 29, 1994**

[51] **Int. Cl.**[6] **H04K 1/00**

[52] **U.S. Cl.** **380/21**; 380/23; 380/25; 380/49

[58] **Field of Search** 380/23, 24, 25, 380/4, 21, 49

[56] **References Cited**

U.S. PATENT DOCUMENTS

4,238,854	12/1980	Ehrsam et al. .	
4,652,990	3/1987	Pailen et al. .	
5,081,678	1/1992	Kaufman et al. .	
5,349,643	9/1994	Cox et al.	380/25

OTHER PUBLICATIONS

W. M. Goode, "Securing Personal Computers In A Network Environment", Micronyx, Inc. 1901 N. Central Expressway, Richardson, Tex.—document 01-214-690-0595, pp. 135-148.

H. Feinstein, "Security On Unclassified Sensitive Computer Systems", Nat'l. Computer Security Conference Proceedings, Sep. 15-18, 1986, pp. 81-90.

S. Cobb, "Security Software", Which Computer, Sep. 1991, pp. 64-75.

J. G. Steiner, "Kerberos: An Authentication Service For Open Network Systems", Presented at Winter USENIX, 1988, Dallas, Texas.

IBM Technical Disclosure Bulletin, vol. 28, No. 12, May 1986 "Integrity Of Stored Public Key", pp. 5168-5169.

Primary Examiner—David C. Cain

Attorney, Agent, or Firm—Daniel P. Morris

[57] **ABSTRACT**

A computing system is described having an automated management system for managing keys to encrypt and decrypt stored data on the computing system. The computing system has an authentication server; a key client; a key generator; a key server; a key database; and an encrypted data file memory. The authentication server authenticates the user and in response to the user accessing the computing system the authentication server provides the user with a ticket validating the user. The key client of a creating user when creating a data file invokes the generator to generate a key corresponding to the data file. The key is provided to the key server and the key client uses the key to encrypt the data file which is stored in the encrypted data file memory. The key client of an accessing user sends its ticket and data file identification data to the key server. The key server checks the ticket and sends the key corresponding to the data file to the key client of the accessing user. The key client of the accessing user uses the key to decrypt the encrypted data file. The stored data can further include a header containing the key and owner and permitted user identification data. The ticket can contain a key to encrypt messages sent between the client server and key client.

39 Claims, 4 Drawing Sheets

<table>
<tr><td rowspan="2">LENGTH OF FILE HEADER</td><td rowspan="2">FILE ENCRYPTON KEY, ITSELF ENCRYPTED UNDER A CONTROL KEY</td><td rowspan="2">CONTROL KEY INDEX NUMBER</td><td rowspan="2">FILE OWNER'S NAME</td><td colspan="2">ACCESS CONTROL LIST</td><td rowspan="2">MESSAGE AUTHENTICATION CHECK</td></tr>
<tr><td>NAME OF ACCESSOR 1</td><td>• • •</td></tr>
</table>

FIG. 1

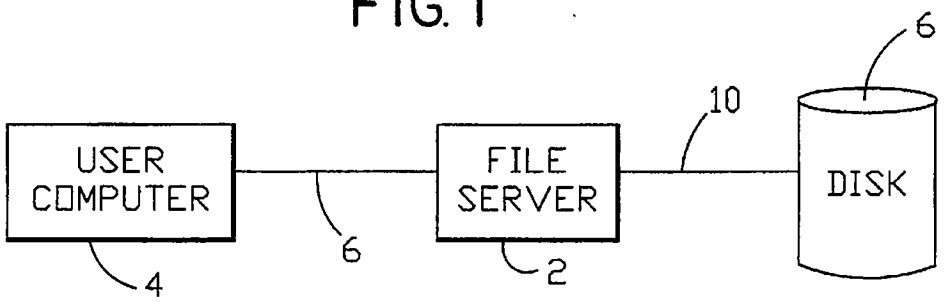

6
10
USER COMPUTER
FILE SERVER
DISK
6
4
2

FIG. 2

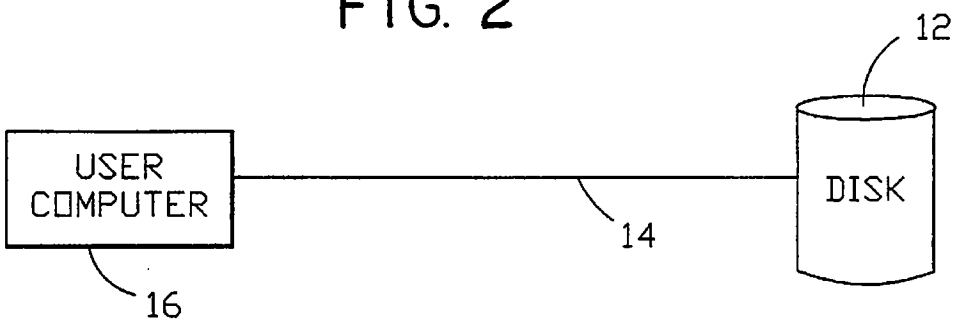

12
USER COMPUTER
DISK
14
16

FIG. 3

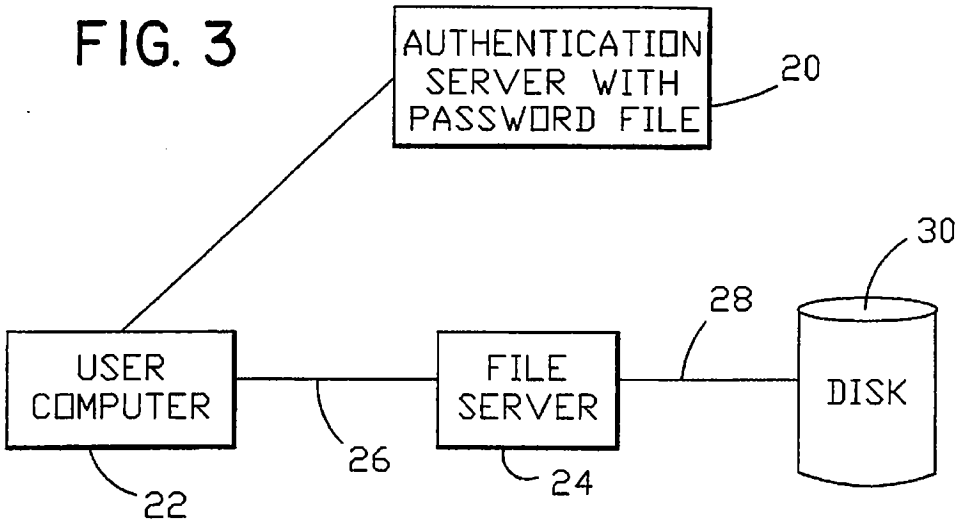

AUTHENTICATION SERVER WITH PASSWORD FILE
20
30
28
USER COMPUTER
FILE SERVER
DISK
26
22
24

FIG. 4

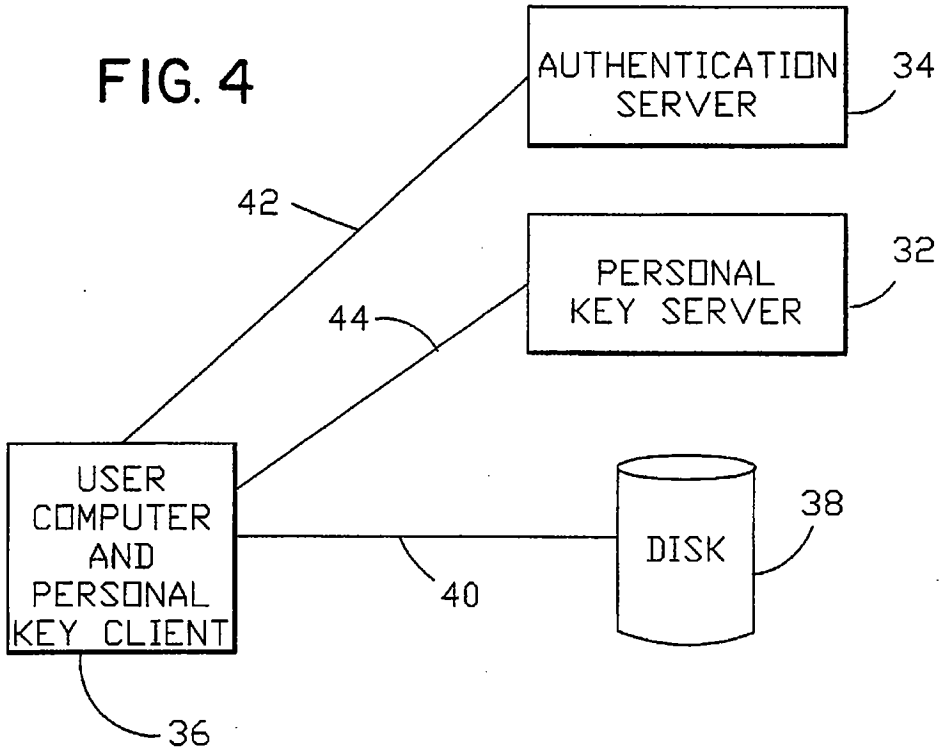

AUTHENTICATION SERVER
34
42
PERSONAL KEY SERVER
32
44
USER COMPUTER AND PERSONAL KEY CLIENT
DISK
38
40
36

FIG. 5

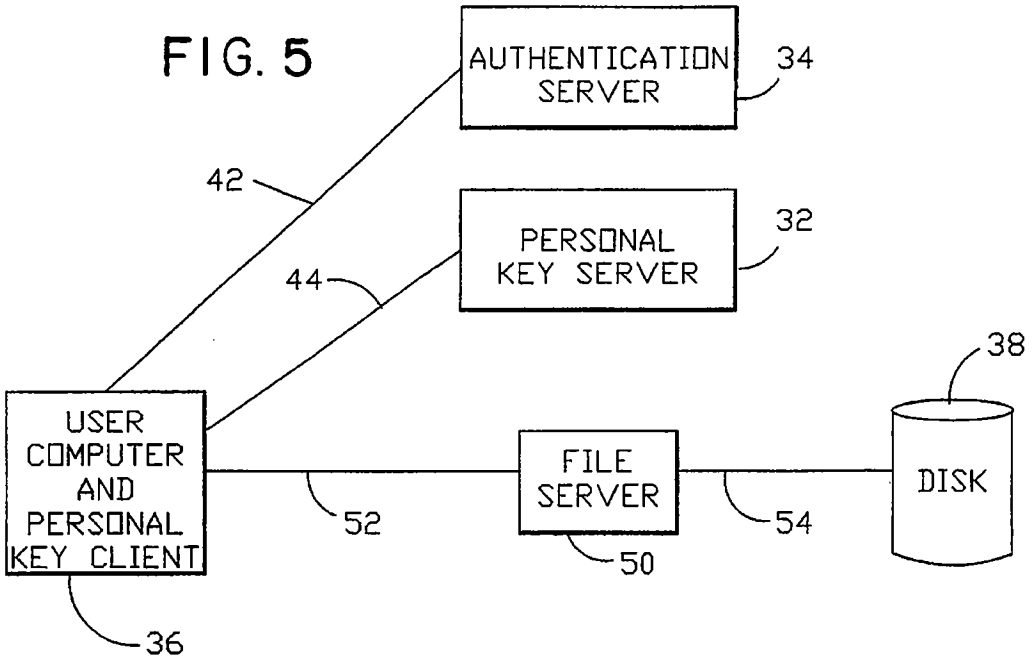
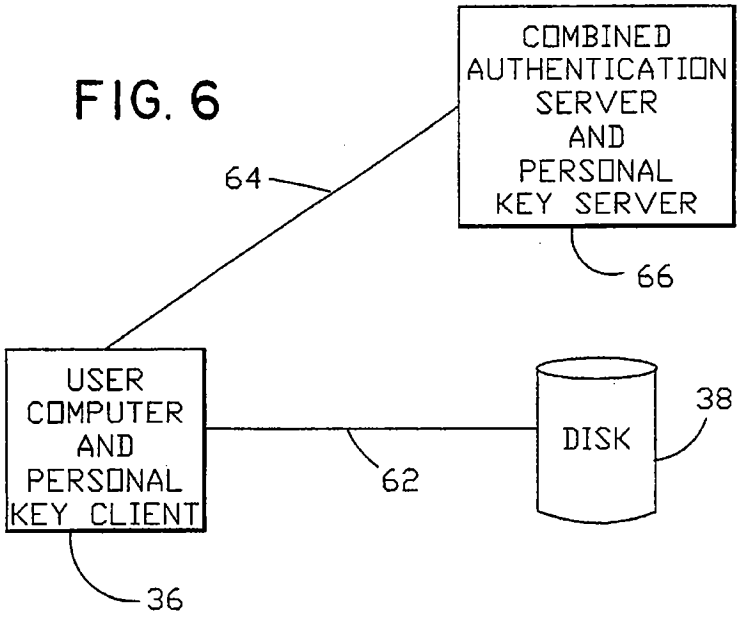

AUTHENTICATION SERVER
34
42
PERSONAL KEY SERVER
32
44
38
USER COMPUTER AND PERSONAL KEY CLIENT
FILE SERVER
DISK
52
54
50
36

FIG. 6


COMBINED AUTHENTICATION SERVER AND PERSONAL KEY SERVER
64
66
USER COMPUTER AND PERSONAL KEY CLIENT
DISK
38
62
36

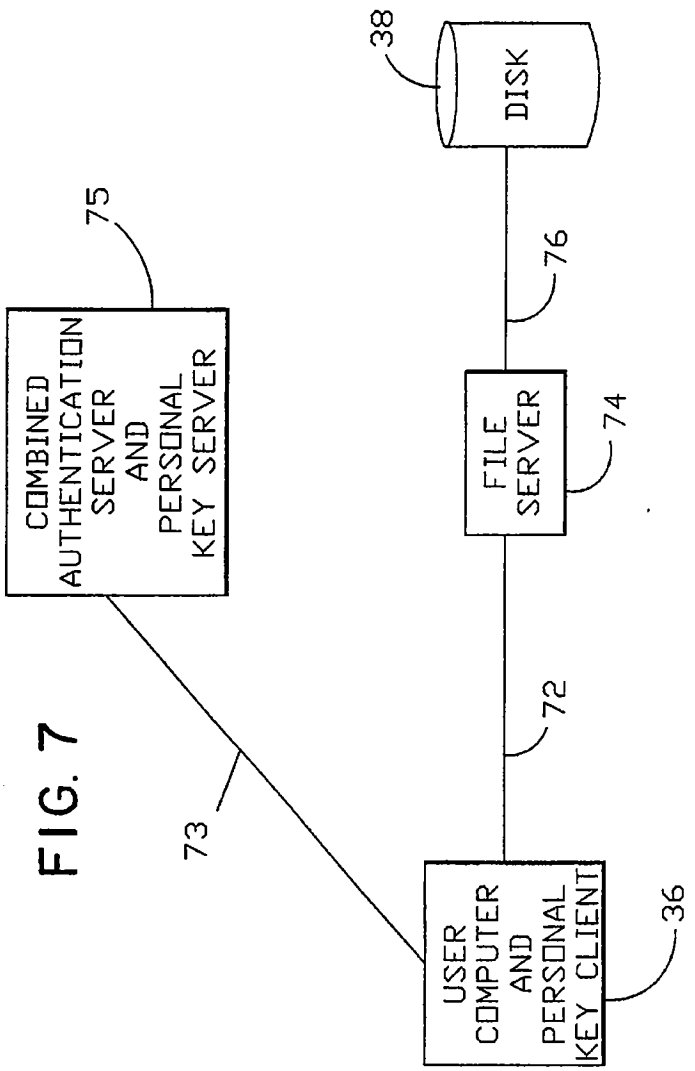

COMBINED AUTHENTICATION SERVER AND PERSONAL KEY SERVER
75
DISK
38
76
FILE SERVER
74
72
73
USER COMPUTER AND PERSONAL KEY CLIENT
36

FIG. 7

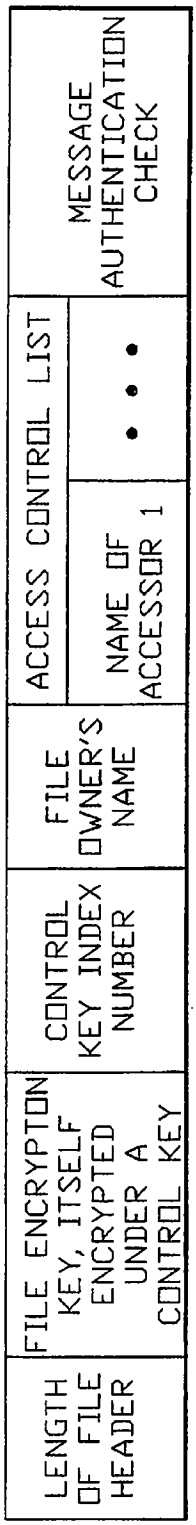

LENGTH OF FILE HEADER
FILE ENCRYPTION KEY, ITSELF ENCRYPTED UNDER A CONTROL KEY
CONTROL KEY INDEX NUMBER
FILE OWNER'S NAME
ACCESS CONTROL LIST
NAME OF ACCESSOR 1
• • •
MESSAGE AUTHENTICATION CHECK

FIG. 8

PERSONAL KEY ARCHIVE

FIELD OF THE INVENTION

This invention relates generally to the field of data processing systems. In particular, this invention relates to a security system for a networked data processing system. More specifically, this invention relates to a multiple workstation networked data processing system in which limited access to secure files on a secure workstation is granted to a protected class of users.

BACKGROUND OF THE INVENTION

The power of computing has grown, and continues to grow, rapidly. This increased computing power has provided users of the computing power new opportunities to use computers in new ways. Specifically, computing systems have evolved from being a large central process unit (CPU) with multiple terminals, to being multiple smaller processors interconnected with each other in a distributed processing environment. This shift in processor distribution altered the security problems involved with maintaining limited access to specified data in the processing system. When the computer system was a large CPU with many terminals, access to data was controlled through the operating system of the one CPU which allocated the resources of the CPU. This meant that controlling access to data was relatively simple through the technique of using passwords to identify the user. In a distributed processing environment however, a single processor does not have the resources to identify all users so that password identification is not practical. Moreover, there is no central control over the workstations so that security systems based on an operating system must be replicated in each workstation which is an inefficient use of the processor resources.

Data encryption is a term used for a method of preserving the privacy of data stored in a computing system or communicated over a network. For example, the Data Encryption Standard (reference 2) defines a method of encrypting data, and the IBM Information Protection System (reference 3) applies that standard to computer files. The latter product requires users to manually invoke encryption for specific files, whereas the Cryptographic File System for Unix (reference 1) automatically applies cryptography to files. In each of these products, an encryption key must be manually supplied by the user. The present invention provides a way to automate the handling of such encryption keys.

There are two basic types of data encryption methods: conventional or symmetric methods, such as DES reference 2, and public-key or asymmetric methods such as RSA reference 7. Conventional methods use the same key for both encrypting and decrypting data; public-key methods use different keys for the two operations. Conventional methods are generally faster than public-key encryption, and are thus more appropriate for bulk data file encryption as envisaged in this invention. The principal disadvantage of conventional encryption is complexity in the management of encryption keys. The purpose of the present invention is to ameliorate this disadvantage by providing a way to manage encryption keys used for conventional encryption of data files.

Message authentication is a term used for any procedure for "... determining with a high level of confidence whether a string of text (plaintext or ciphertext) has been altered (accidentally or intentionally)" (reference 5, p. 100). Message authentication should not be confused with user authentication and network authentication, which are described below. Message authentication can be used to verify the integrity of the contents of data files stored in computer systems. Data files that are protected with message authentication techniques can be themselves stored in either encrypted or plaintext form.

Procedures for message authentication are described in pages 100–105 and 359–367 of reference 5. These procedures depend upon the use of a key for encrypting a message authentication check. The present invention system provides a way for automating the management of such keys.

Many data encryption and message authentication systems require users to manually provide encryption keys. These keys are required both when files are first encrypted and later when they are decrypted. Disadvantages of manual key management include the awkward and time-consuming requirement for end-users to enter encryption keys, the possibility that users may forget keys, the likelihood that users may select cryptographically weak keys, the inability to access encrypted files when the individual who knows the keys is unavailable, and the need to distribute keys to all individuals who share access to encrypted files. The system according to addresses these issues by providing a way to automatically manage encryption keys.

In recent years, the reduced cost of computing equipment has encouraged the use of large numbers of small computers. Often, these are interconnected via computer networks to form distributed systems with many interdependent functions. For a distributed system shown in FIG. **1**, data files on disk **8** may be transferred over link **10** and stored on a file server **2** that is remote from the computer **4** that access the files on the file server **2** through network **6**. This is an alternative to the more traditional local storage of data files on disk **12** directly connected by link **14** to the computer **16** that access the files, as shown in FIG. **2**.

Note that when data files are stored on a file server **2**, as in FIG. **1**, the data traverses a computer network **6** that is, in many cases, shared with many other user computers. In this situation, it is generally technically easy for equipment connected to the network to read the data bytes as they are transmitted between the computer user and the file server. Encrypting the data is necessary if privacy is desired. However, encryption requires some method to coordinate the encryption keys used for communication on the network.

A file server **2**, as shown in FIG. **1**, is often shared by multiple user computers for two reasons: (1) to amortize the cost of the file server over many users; and (2) to permit users to share data among themselves. Typically, the file server provides access controls which permit file owners to specify which other users can share their files. For example, user A may indicate that user B may read file **1** while user C may read and write file **2**.

Although file access controls are effective for limiting the access of end-users to each others' files, access controls do not ensure complete privacy of files. Typically, system administrators have the ability to override these controls for purposes such as performing file backup. Data encryption of files has the advantage that only users who have the correct encryption keys can make use of the contents of files.

File access controls imply that the file server is able to reliably and securely identify users who request access to files. Typically, users identify themselves by executing a login process that involves entering a computer userid and matching password. The mechanism For validating the userid and password, and for maintaining the connection between the user and any processes run on behalf of the user,

is called user authentication. User authentication should not be confused with message authentication, as discussed above. For example, referring to FIG. 1, the userid and password may be checked against a password file in either the client computer 4 or the file server 2, or both. Note that the password must be transmitted across the network 6 if the password validation is performed within the file server 2. Hence, the password itself should be encrypted if there is a concern about network eavesdropping.

A network authentication mechanism, such as Kerberos (reference 8), keeps the password file on a authentication server 20 as shown in FIG. 3. A special protocol is used to validate a userid and password entered on a user computer 22 against the password file on the authentication server 20. The latter generates authentication data, embodied in a ticket, that identifies the user. For example, the user computer obtains from the authentication server 20 a ticket to access the file server 24. The user computer 22 forwards this ticket to the file server 24 whenever the user wants to accesses a file. The file server 24 relies on the contents of the ticket to identify the user. The files are retrieved over link 28 from disk 30 to file server 24.

Kerberos uses cryptographic techniques to avoid sending the password on the network 26, to protect the contents of tickets, and to allow the file server 24 to be certain that the tickets are both valid and issued by the authentication server 20. Advantages of this scheme include (1) the password is kept in one place rather than in (potentially) multiple user computers or file servers 24; (2) the password is not transmitted over the computer network 26; (3) each ticket contains a dynamically-generated encryption key shared by the user computer 22 and the file server 24.

Other mechanisms can be used for network authentication. For example, KryptoKnight (reference 6), uses somewhat different protocols to achieve functions similar to Kerberos. As another example, a network authentication mechanism could be based upon public-key cryptography.

The previous discussion is presented in terms of file service, but applies equally to database service (both local to a user computer or distributed to a database server). The same mechanisms for user authentication and data access control are used with database systems. As with file services, complete data privacy can be achieved in database systems only by applying data encryption techniques. The latter imply the need to manage data encryption keys.

REFERENCES

1. Blaze, Matt, A Cryptographic File System for Unix, 1st ACM Conference on Communications and Computing Security, Fairfax, Va., Nov. 3–5, 1993.

2. Data Encryption Standard, National Bureau of Standards, Federal Information Processing Standards Publication Number 46, National Technical Information Service, Springfield, Va., Jan. 15, 1977.

3. IBM, Information Protection System Cryptographic Programs for CMS (IPS/CMS) User's Guide, IBM order number SH20-2621.

4. Kazar, Michael Leon, Ubik: Replicated Servers Made Easy, Proceedings of the Second Workshop on Workstation Operating Systems, Sep. 27–29, 1989.

5. Meyer, Carl H., and Matyas, Stephen M., Cryptography: A New Dimension in Computer Data Security, John Wiley & Sons, New York, 1982.

6. Molva, R., Tsudik, G., Van Herreweghen, E., Zatti, S., KryptoKnight Authentication and Key Distribution Service, Proceedings of ESORICS 92, Toulouse, October, 1992.

7. Rivest, R. L., Shamir, A., and Adleman, L., A Method for Obtaining Digital Signatures and Public-Key Cryptosystems, Communications of the ACM, Volume 21, Number 2, pages 120–126 (1978).

8. Steiner, Jennifer G., Neuman, B. Clifford, and Schiller, Jeffrey I., Kerberos: An Authentication Service For Open Network Systems, Usenix Conference Proceedings, pages 183–190, February 1988.

9. Howard, John L.; Kazar, Mitchael L.; Menees, Sherri G.; Nichols, David A.; Satyanarayanan, M.; Sidebotham, Robert N.; West, Michael J.; Scale and Performance in a Distributed File System, ACM Transactions on Computing Systems, Vol 6, No. 1, February 1988, pp 51–81.

10. Sandberg, Russel; Goldberg, David; Kleiman, Steve; Walsh, Bob; and Lyon, Bob, Design and Implementation of the SUN Network File System, USENIX Summer Conference Proceedings, Summer, 1985, pp 119–130.

OBJECTS OF THE INVENTION

It is an object of the present invention to provide an improved computing system.

It is another object of the present invention to provide an improved computing system having improved security.

It is a further object of the present invention to provide an improved computing system having improved security in a distributed computing environment.

It is still another object of the present invention to provide an improved computing system having improved security in a distributed computing environment in which system maintenance may be performed despite the non-availability of the file passwords.

It is another object of the present invention to provide an automatic system for managing keys used to encrypt data files stored in distributed computing systems.

It is an additional object of the present invention to provide an automatic system for managing encryption keys used to ensure the integrity of data files stored in distributed computing systems.

It is an additional object of the present invention to ensure the privacy of the encryption keys; that is, to ensure that only authorized persons or processes are permitted to retrieve encryption keys.

It is an additional object of the present invention to provide a way to recover encryption keys when authorized persons are unavailable (such as when employees have departed).

It is another object of the present invention to permit system management functions, such as file backup, to operate in a normal manner without knowledge of the encryption keys.

It is another object of the present invention to minimize technical costs (such as database size) and complexity (such as number of administrative functions) associated with the automatic key management system.

SUMMARY OF THE INVENTION

A broad aspect of the present invention is a a computing system having a security system for identifying if a user is permitted to create or access a data file on the computing system. The computing system has an authentication server; a key client; a key generator; a key server; a key database; an encrypted data file memory; the authentication server authenticates the user as permitted accessing the computing

system the authentication server provides the user with a ticket validating the user as permitted to operate on the computing system; the key client of a creating user when a creating user creates a data file invokes the generator to generate a key corresponding to the data file; the key is provided to the key server; the key client of the creating user uses the key to encrypt the data file to form an encrypted data file which is stored in the encrypted data file memory; the key client of an accessing user, when an accessing user accesses the data file, sends the ticket and said data file identification data to the key server; the key server checks the ticket to verify that the accessing user is permitted to access the data file; the key server sends the key corresponding to the data file to the key client of the accessing user; and the key client of the accessing user uses the key to decrypt the encrypted data file.

Another broad aspect of the present invention is a method for providing a security system for identifying if a user is permitted to create or access a data file on the computing system. The method includes the steps of authenticating said user as permitted to operate on said computing system and in response to said user accessing said computing system said user is provided with a ticket by an authentication validating said user as permitted to operate on said computing system; when a creating user creates a data file said creating user invokes a generator to generate a key corresponding to said data file, said key is provided to said key client, said key client of said creating user uses said key to encrypt said data file to form an encrypted data file which is stored in an encrypted data file memory; said key client of an accessing user, when an accessing user accesses said data file, sends said ticket and said data file identification data to said key server, said key server checks said ticket to verify that said accessing user is permitted to access said data file, said key server sends said key corresponding to said data file to said key client of said accessing user, said key client of said accessing user uses said key to decrypt said encrypted data file.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other objects, features and advantages of the present invention will become apparent upon a consideration of the following detailed description of the invention when read in conjunction with the drawing Figures, in which:

FIG. **1** shows a schematic diagram of a distributed system having a file server which stores data remote from a user computer.

FIG. **2** shows a schematic diagram of a system using data stored locally to a user computer.

FIG. **3** shows a schematic diagram of Kerberos network authentication system.

FIG. **4** shows a schematic diagram of a Personal Key Archive security system according to the present invention with files stored on a local disk and a separate personal key server.

FIG. **5** shows a schematic diagram of another Personal Key Archive security system according to the present invention with files on a file server and a separate personal key server.

FIG. **6** shows a schematic diagram of another Personal Key Archive security system according to the present invention with files stored on a local disk and the personal key server co-located with the network authentication server.

FIG. **7** shows a schematic diagram of another Personal Key Archive security system according to the present invention with files stored on a file server and the personal key server located in the network authentication server.

FIG. **8** shows the organization of a file header used with the Personal Key Archive according to the present invention.

DETAILED DESCRIPTION

The security system, according to the present invention, automatically manages keys used for encryption or message authentication of data files or individual entries in databases. The files or databases may be stored on servers **2**, as in FIG. **1**, or on user computers **16**, as in FIG. **2**. In the preferred embodiment of the present invention uses a distributed environment in which user computers and various servers are interconnected via a communications network. The invention uses a network authentication mechanism.

The personal key archive security system according to the present invention uses two new components as shown in FIGS. **4** through **7**: the Personal Key Client component on user computer **36** and the Personal Key Server. The Personal Key Server component **32** may execute upon a separate computer, as shown in FIGS. **4** and **5**, or may be co-located with the authentication server portion as in FIGS. **6** and **7**. For effective security, the Personal Key Server **32** is preferably not be located on the same machine as the data files themselves which are stored on local disc **38**. Servers useful to practice the present invention are described in references **9** and **10** above.

Referring to FIG. **4** data files arc stored on local disc **38** which is connected to user computer **36** by network **40**. Authentication server **34** is connected to user computer **36** by network **42**. Personal key server **32** is connected to user computer **36** by network **44**. Although FIG. **4** shows one user computer **36**, it will be understood that there can be a plurality of user computers each connected to the authentication server **34** and Personal Key Server **32** by network corresponding to **42** and **44**. These same comments apply to the embodiments of FIGS. **5–7**.

The type of Server in FIG. **4** is a Key Distribution Facility. All user computers are able to communicate with Personal Key Server **32**. The Personal Key Server **32** is trusted, meaning that all user computers **36** expect that the Personal Key Server **32** will perform its function reliably and securely. The Personal Key Server is physically and logically secure against intrusion by anybody other than administrative staff. If required by an installation, the Personal Key Server may be replicated on multiple computers, using the techniques described in reference **4**, in order to improve operational reliability.

FIG. **5** schematically shows another Personal Key Archive according to the present invention wherein all elements of FIGS. **4** and **5** identified with the same reference numeral correspond to the same thing. The personal key archive of FIG. **5** includes a file server **50** connected to user computer **36** by network **52** and connected to disk **38** by link **54**.

FIG. **6** schematically shows another Personal Key Archive system according to the present invention. Local user computer **38** includes a personal key client and data files are stored on a local disk **36** which is connected to local user computer **36** by link **62**. Local computer **36** is connected by network **64** to a combined authentication server and personal key server **66**.

FIG. **7** shows another schematic diagram of another embodiment of a Personal Key Archive security system according to the present invention. A user computer **36**

containing Personal Key (client is connected by network **72** to file server **74** which stores data fields. File server **711** is connected by link **76** to disk **38**. The user computer **36** is connected by network **73** to combined authentication servers and Personal Key Server **75**.

In any configuration, the Personal Key Server maintains a Personal Key Database that contains certain information required to decrypt files or check their message authentication. The Personal Key Client communicates with the Personal Key Server as files are created and accessed. The Personal Key Server matches information in the Personal Key Client's messages with the contents of the database to determine the appropriate response to the Personal Key Client.

The Personal Key Archive provides two mechanisms for automatic key management: a basic method, and an enhanced method. The basic mechanism is discussed in order to clearly describe the underlying concept. The enhanced method offers certain advantages that are described below. Both methods can be implemented in any of the configurations shown in FIGS. **4** through **7**, and both methods can manage keys used For either data encryption or message authentication, or both. Both methods can be applied to entire data files or to individual records in database systems. This description generally discusses the encryption or message authentication of entire data files. However, it should be understood that both the basic and enhanced methods can also be applied to individual records in databases.

In the basic method, each data file is encrypted by the Personal Key Client, on the user's computer, using a randomly-chosen key generated by the Personal Key Server at the time the file is created. The key is stored in the Personal Key Database located on the Personal Key Server. Kerberos or KryptoKnight tickets are used to identify the user to the Personal Key Server when the file is created or accessed. Although Kerberos and KryptoKnight are used herein as a source of the ticket, this is exemplary only and not limiting.

The Personal Key Server Database contains an entry for each file that is encrypted. These entries are indexed by information that identifies the files, such as the names and creation dates of the files. Each entry contains the key used to encrypt the corresponding file, the name of the owner of the file, and an access control list containing the names of any other users who are permitted to access the file.

When a file is created, the Personal Key Client sends the Kerberos ticket of the creator, along with the file's name and creation date, to the Personal Key Server. The Personal Key Server randomly generates a file encryption key, creates a new entry in the database, and responds to the Personal Key Client with the file encryption key. The Personal Key Client then uses the key to encrypt the data as it is written to the file.

When a file is accessed (both for reading and for updates), the Personal Key Client sends the Kerberos ticket of the accessor, the file's name, and the file's creation date to the Personal Key Server. The latter retrieves the appropriate entry in the database and checks the identity of the accessor (as provided in the Kerberos ticket) against the file owner's name and the access control list in the database entry. If the accessor is either the owner or one of the users named in the access control list, the Server sends the file encryption key back to the Personal Key Client. The latter uses the key to decrypt the data as it is read from the file.

Certain user actions require that additional messages be exchanged with the Personal Key Server in order to maintain the accuracy of the Personal Key Database:

- When a file is renamed (or any other identifying information, such as the file creation date, is changed) the Personal Key Client preferably sends the new and old filenames (or other identifying data) to the Personal Key Server so that the Personal Key Server can update the Personal Key Database.
- If the recorded owner of a file changes, the Personal Key Server is preferably notified so that the appropriate Personal Key Database entry can be updated.
- If there are modifications to the access control list messages are preferably sent to the Personal Key Server to cause updates to the Personal Key Database.

These user actions may only be performed by the owner of the file, as identified in the Personal Key Database entry and verified via the Kerberos ticket accompanying the messages between the Personal Key Client and Personal Key Server.

The messages sent between the Personal Key Client and the Personal Key Server are themselves encrypted in session keys that are provided by Kerberos. This "double encryption" ensures that the file encryption keys themselves do not appear in the clear on the communication path between the Client and the Server.

The basic mechanism can be applied to protect individual records in database systems by indexing the entries in the Personal Key Database by the same indexing information used for accessing the database records themselves.

The basic mechanism can be adapted to verify the integrity of files or database records against inadvertent or malicious modifications by adding an encrypted message authentication check field to the end of the files or records, as described on page 100 of reference 5. This field is encrypted under a key managed by the basic mechanism as described above. The field is generated when a file is written, and is verified either by completely scanning the file or record when it is opened, or alternatively verified only when a file is fully read by an application program.

A minor variation of this method is to have the Personal Key Client (rather than the Server) generate the file encryption key. This would be sent to the Personal Key Server at the time a file is created, so that the Personal Key Server could store the key in the Personal Key Database. A disadvantage of this variant is that it would be harder to ensure the cryptographic quality of the key if generated at the Personal Key Client compared to generating the key at the Personal Key Server.

The enhanced method combines a technique described on page 283 of reference 5 with the basic method described above. The header associated with each file preferably accompanies the file should the file be moved or renamed or backed-up. The header may be associated with the file in a number of ways: as the first few bytes of the file, or as a trailer at the end of the file, or stored in the file's directory entry, or in a separate area associated with the file's directory entry, or in a local database. These ways are listed as examples of methods of associating the files and the headers; other methods may also be used.

An example of a file header is shown in FIG. **8** and contains the file encryption key for the file, itself encrypted under a control key (defined below). The header also contains the control key index number, the name of the owner of the file, and the access control list of users permitted to access the file. The entire file header is "protected" against modification by a message authentication check field that is appended to the header and is encrypted under the same control key.

The control key mentioned above is a randomly-generated key that is used only to encrypt the individual file encryption

keys of multiple files, and to encrypt the message authentication check fields of the headers of the same files. One control key is used for multiple files; a new control key is generated for every N files and/or when M hours or days have elapsed. N and M are parameters that could be varied for different implementations or installations. The fact that individual control keys are shared by multiple files is not a cryptographic weakness, since the keys are used only to encrypt file encryption keys that are themselves random numbers.

Control keys are generated by and kept entirely within the Personal Key Server. At any given time, the control key currently being used for newly-generated files is called the current control key. Outside the Personal Key Server, control keys are identified by control key index numbers, which are unique numbers that identify individual control key values. Each file header contains the index number that identifies the particular control key used to encrypt the file encryption key contained in the same header.

In this enhanced method, the Personal Key Database is structured very differently from the basic method. The Database contains an entry for each control key that has been generated. The database entries are indexed by the control key index numbers. They contain the actual values of the control keys themselves.

The operation of the enhanced method is similar to the basic method. Each data file is encrypted by the Personal Key Client, on the user's computer, using a randomly-chosen key generated by the Personal Key Server at the time the file is created. Kerberos or KryptoKnight tickets are used to identify the user to the Personal Key Server when the file is created or accessed. Unlike the basic method, the file encryption key is stored in the file header, and is kept from public use by encrypting the file encryption key itself under a control key known only to the Personal Key Server.

When a file is created, the Personal Key Client sends a message to the Personal Key Server with a Kerberos ticket identifying the file creator. The Personal Key Server prepares a file header as outlined above, and sends the header and the file encryption key itself back to the Personal Key Client. The latter stores the file header with the file, and uses the key to encrypt the file as it is written by the application.

When a file is accessed (both for reading and for updates), the Personal Key Client reads the file header and sends it to the Personal Key Server, along with a ticket identifying the accessor. The Personal Key Server uses the control key index number in the header to lookup the control key in the Personal Key Database. The Personal Key Server then uses the control key to validate the message authentication check field; if it is invalid, the Personal Key Server rejects the access request. If the header is valid, the Personal Key Server then compares the accessor's name (from the ticket) against the name of the file owner and the names in the access control list. If the accessor's name is not found, then the Personal Key Server rejects the access request. If the name is found, then the Personal Key Server decrypts the file encryption key and sends it back to the Personal Key Client. The Personal Key Client can then use the file encryption key to decrypt the file as it is read.

When a file owner wishes to add or delete a user from the access control list, or when the file owner's name is changed, the Personal Key Client sends the file's current header to the Personal Key Server, along with the details of the change. The Personal Key Server validates the header in the same manner as when a file is accessed, and then updates the header as requested by the file owner. The Personal Key Server then sends the updated header back to the Personal Key Client, which inserts the updated header into the file.

The messages sent between the Personal Key Client and the Personal Key Server are themselves encrypted in session keys that are provided by Kerberos. This "double encryption" ensures that the file encryption keys themselves do not appear in the clear on the communication path between the Personal Key Client and the Personal Key Server.

The enhanced mechanism can be applied to protect individual records in database systems by extending such records with a new field containing the header shown in FIG. 8. The header then applies to the particular database record. The operations of creating, accessing, and controlling access to the records operate in a manner similar to that described above.

The enhanced mechanism can be adapted to verify the integrity of files or database records against inadvertent or malicious modifications by adding another message authentication check field to the header. This second message authentication check applies to the contents of a file or record, rather than to the header. This second check is generated when the file is closed, and is validated either by scanning the file or record when it is opened, or by verifying the check when the file is fully read by an application.

As specified above, a single control key is used for two purposes: to encrypt the file encryption key and to encrypt the message authentication field. At very minor additional cost, additional control keys or other values could be stored in the Personal Key Database. Such values might include a secondary control key for the message authentication check of the header, another control key for the message authentication check of the file itself, and/or an initialization vector for use when encrypting the file. These values would all be referenced by the single control key index number stored in the header of each file.

As with the basic mechanism, a minor variation of this method is to have the Personal Key Client (rather than the Personal Key Server) generate the file encryption key. This would be sent to the Personal Key Server at the time a file is created, so that the Personal Key Server could generate the file header. One disadvantage of this variant is that it would be harder to ensure the cryptographic quality of the key if generated at the Personal Key Client compared to generating the key at the Personal Key Server. A more serious disadvantage of this variant is that it would permit a modified Personal Key Client to attempt discovery of the current control key by using chosen file encryption keys in a "chosen plaintext" type of cryptographic attack.

The two mechanisms share a number of advantages:

1. File encryption keys are automatically managed.

a. No additional effort is required on the part of users.

b. Only persons who have the Kerberos tickets of the owners of files, and other users given access by the file owners, can read the unencrypted form of the files.

c. When a file is shared among multiple users, its file encryption key is automatically made available to all users.

d. File management functions, such as backup and restoration of files, operate without change. Persons who control such functions can view only the encrypted form of the files; they have no access to the unencrypted versions of the files. (This assumes that personnel who perform file management functions are distinct from those who administer the Kerberos authentication database. As discussed below, the latter can always gain access to encrypted files by changing the passwords of Kerberos accounts.)

e. Users cannot forget or lose file encryption keys. Hence there is no risk of inadvertent loss of access to data files.

f. Each file is encrypted under a unique key. The fact that the keys are unique tends to frustrate certain types of cryptographic analysis of the encrypted files.

g. File encryption keys are randomly generated, making them of higher cryptographic quality than if they were chosen by users.

2. The "foundation" for access to files is the Kerberos or KryptoKnight authentication of individual users. This has several consequences:

a. Typically, Kerberos implementations depend upon passwords, but they could also use other means of identifying individuals such as smart cards. Personal Key Archive can work with and enhance an installation that uses any method of validating users to Kerberos.

b. A user can change his or her Kerberos password without any impact on the encrypted data files or the contents of the Personal Key Database.

c. If a user leaves an installation, the organization can recover access to the user's encrypted files by resetting the user's Kerberos password. Hence the organization has no risk of inadvertent loss of access to data files due to the unavailability of file owners.

d. A corollary of the previous item is that the security of encrypted files is equivalent to the security of the Kerberos authentication server; whoever can update the Kerberos database can access all encrypted files. This is why it is acceptable to co-locate the Personal Key Server and the Kerberos server as shown in FIGS. **6** and **7**.

3. In file server configurations (as in FIG. **1**), files are encrypted on user computers rather than on the file server. This has several beneficial aspects:

a. The computing load of data encryption is distributed among multiple user machines rather than concentrated on a single, shared file server. Hence the total network capacity for data encryption increases as more user computers are added to the network.

b. The computing load of data encryption is incurred by the user machines. Hence any economic tradeoff between performance and machine capacity can be customized to the needs of individual users.

c. Data transmitted over the network between user machines and the file server is encrypted. Hence it cannot be read by tapping the network.

The basic method described above has several serious limitations:

1. The size of the Personal Key Database is related to the total number of files in the installation. That could easily amount to tens or hundreds of thousands of entries. Furthermore, the individual entries must be varying-length in order to accommodate varying numbers of entries in the access control lists. Furthermore, these lists can be updated at any time, so the entire Personal Key Database must be read-write. The combination of a large database with varying-length read-write entries makes For a fairly complex database design.

2. The Personal Key Database is preferably backed-up or replicated at least as often as the encrypted files are backed-up, since the loss of the Personal Key Database implies loss of access to the unencrypted form of the files.

3. The Personal Key Database must be updated whenever a file is renamed. In current systems, this may be difficult or impossible to ensure since there are ways that files can be implicitly renamed (e.g. by restoration from a backup tape under another name) without explicit use of the system rename function.

The enhanced method addresses the limitations of the basic method:

1. The number of entries in the Personal Key Database is related to the number of control keys, rather than the total number of data files. If a new control key is generated every N files, then the number of Personal Key Database entries is a fraction (one-Nth) of the number of files. If a new control key is generated every M days, then the number of Personal Key Database entries is related only to the amount of time that has elapsed since the Personal Key Database was created. In the latter case, the total size of the Personal Key Database can be completely estimated in advance.

The size of Personal Key Database entries is fixed, since they contain only the control key and perhaps a few other cryptographic variables. Furthermore, existing entries in the Personal Key Database are never updated; the only changes are the appending of new entries to the end of the Personal Key Database. Hence the design of the Personal Key Database is technically simpler, and the total size of the Personal Key Database is smaller, than with the basic method.

2. The Personal Key Database needs to be backed up or replicated only when a new control key is generated. If the design choice is made to produce a new control key only as time elapses, then the backup or replication schedule can be completely scheduled in advance.

3. Renaming, backing-up and restoring, or other manipulations of the name or location of an encrypted file have no impact on the ability to recover the unencrypted form of the file as long as the header remains associated with the file.

While the present invention has been shown and described with respect to specific embodiments, it will be understood that it is not thus limited. Numerous modifications, changes and improvement will occur which fall within the scope and spirit of the invention.

We claim:

1. A computing system for automatically managing keys to encrypt and decrypt stored data; comprising:

an authentication server;

a key client;

a key generator;

a key server;

a key database;

an encrypted data memory;

said authentication server authenticates said user and provides said user with a ticket identifying said user;

said key client of a creating user, when a creating user creates stored data invokes said generator to generate a key corresponding to said stored data to form encrypted stored data, said key is provided to said key server, said key client of said creating user uses said key to encrypt said stored data which is stored in said encrypted data memory;

said key client of an accessing user, when an accessing user accesses said stored data, sends said ticket and identification data for said stored data to said key server, said key server obtains said authentification data from said ticket for said accessing user, said key server sends said key corresponding to said stored data to said key client of said accessing user, said key client of said accessing user uses said key to decrypt said encrypted stored data.

2. A computing system according to claim **1**, wherein said key client sends said encrypted stored data to said encrypted data memory.

3. A computing system according to claim **1**, wherein said key server stores said key and said file identification data in an entry in a key data base.

4. A computing system according to claim 1 wherein said key is a random number.

5. A computing system according to claim 1, wherein said ticket is a set of authentication data.

6. A computing system according to claim 1, wherein said user is authenticated by said authentication server by providing a userid and password to said authentication server.

7. A computing system according to claim 1, wherein said key client resides on a user computer and said key server is a separate unit from said user computer.

8. A computing system according to claim 7, wherein said authentication server is a separate unit from said key server.

9. A computing system according to claim 7, wherein authentication server and said key server are in the same unit.

10. A computing system according to claim 1, further including a file server.

11. A computing system according to claim 6, further including a file server.

12. A computing system according to claim 7, further including a file server.

13. A computing system according to claim 1, wherein there are a plurality of user computers each having a key client and each connected by a network to said key server and to said authentication server.

14. A computing system according to claim 1, wherein said computing system is a single computing system having a plurality of users and wherein said key client, said authentication server and said key server are parts of said single system.

15. A computing system according to claim 1, wherein said key server distinguishes between said creating user and said accessing user who is not a creating user.

16. A computing system according to claim 15, wherein said key server stores an identification of said creating user, said accessing user and an owning user in said key database.

17. A computing system according to claim 1, wherein there are actions which only a creating user is permitted to perform on said stored data as identified in said key database, messages send between said key client and said key server relating to said actions are accompanied by said ticket of said creating user and said messages are encrypted.

18. A computing system according to claim 17, wherein said actions are selected from the group consisting of renaming said stored data, changing a user who owns said stored data and modifying a list of users permitted to access said data file.

19. A computer system according to claim 1, wherein said generator is part of said key client and said key is sent to said key server when said stored data is created.

20. A computing system according to claim 1, wherein said generator is part of said key server, said key client invokes said generator to generate said key by sending said ticket corresponding to said creating user to said key server, said key server in response to receiving said ticket corresponding to said creating user generates said key, said key server sends said key to said key client of said creating user.

21. A computing system according to claim 1, further including a header associated with said storage data.

22. A computing system according to claim 21, wherein said header accompanies said stored data if said data file is moved, renamed or backed up.

23. A computing system according to claim 21, wherein said header contains said key, an identification of an owner user of said stored data, a message authentication check field, a control key identifier and a list of users permitted to access said stored data.

24. A computing system according to claim 23, wherein said key is encrypted under a control key.

25. A computing system according to claim 24, wherein said control key is a randomly generated key used to encrypt said key of said stored data and to encrypt message authentication check fields of said header.

26. A computing system according to claim 25, wherein said control key applies to N data files where N is greater than or equal to zero.

27. A computing system according to claim 25, wherein said control key applies to said N data files created within a fixed period of time.

28. A method for automatically managing keys used to encrypt and decrypt stored data on a computing system comprising the steps of:

authenticating said user, said user is provided with a ticket identifying said user as permitted to operate on said computing system;

when a creating user creates stored data, said creating user invokes a generator to generate a key corresponding to said stored data, said key is provided to said key client, said key client of said creating user uses said key to encrypt said stored data to form encrypted stored data which is stored in an encrypted data memory;

said key client of an accessing user, when an accessing user accesses said stored data file, sends said ticket and said data file identification data to said key server, said key server checks said ticket to verify that said accessing user is permitted to access said data file, said key server sends said key corresponding to said data file to said key client of said accessing user, said key client of said accessing user uses said key to decrypt said encrypted stored data.

29. A method of modifying stored data on a computer system, comprising:

retrieving encrypted stored data from a first storage media, said encrypted stored data being an encryption of said stored data;

maintaining a database on a second storage media for correlating said encrypted stored data to a userid and a data encryption key;

retrieving said data encryption key corresponding to said encrypted stored data from said second storage media; and decrypting said encrypted stored data using said retrieved encryption key.

30. A computer system, comprising:

means for retrieving encrypted stored data from a first storage media, said encrypted stored data being an encryption of said stored data;

means for maintaining a database on a second storage media for correlating said encrypted stored data to a userid and a data encryption key;

means for retrieving said data encryption key corresponding to said encrypted stored data from said second storage media; and

means for decrypting said encrypted stored data using said retrieved encryption key.

31. A computer system according to claim 1, wherein said stored data is selected from the group consisting of a data file and a data base record.

32. A computing system according to claim 1, further including a means to validate said user as permitted to operate on said computing system.

33. A computing system according to claim 21, further including a means for sending encrypted messages between said personal key client and said personal key server.

34. A computing system according to claim 33, wherein said means for sending encrypted messages encrypts said messages using a message encryption key provided by said ticket.

35. A computing system for automatically managing keys to encrypt and decrypt stored data;

an authentication server;

a key client;

a key generator;

a key server;

a key database;

an encrypted data memory;

said authentication server authenticates said user and provides said user with a ticket identifying said user;

said key client of a creating user, when a creating user creates said stored data invokes said generator to generate a key corresponding to said stored data, said key is provided to said key server, said key client of said creating user uses said key to encrypt said stored data to form an encrypted stored data which is stored in said encrypted data memory;

said key client of an accessing user, when an accessing user access said stored data sends said ticket and stored data identification data to said key server, said key server checks said ticket to verify that said accessing user is permitted to access said stored data said key server sends said key corresponding to said stored data to said key client of said accessing user, said key client of said accessing user uses said key to decrypt said encrypted stored data;

a header associated with said stored data;

said header contains said key, an identification of an owner of said stored data, a message authentication check field, a control key identifier and a list of user permitted to access said stored data;

said key is encrypted under a control key;

said control key is used to encrypt message authentication check fields of said header;

said ticket contains a message key for encrypting messages sent between said key client and said key server.

36. A computing system according to claim 1, wherein said computing system is an automated management system for automatically managing keys to encrypt and decrypt stored data.

37. A computing system according to claim 3, wherein said key server stores an identification of said creating user, said accessing user and an owning user in said key database.

38. A method of modifying stored data on a distributed computer system, comprising:

authenticating an identity of a user via an authentication system that provides identification tickets, said user having a userid;

storing encrypted data on a first storage media, said encrypted data being an encryption of said stored data;

maintaining a database on a second storage media for correlating said encrypted stored data to a data encryption key and said userid;

validating said userid identified in an authentication ticket against said userid contained in said database, and automatically choosing whether to grant access to said encrypted stored data;

retrieving said data encryption key corresponding to said encrypted stored data from said second storage media; and

decrypting said encrypted stored data using said retrieved encrypted key.

39. A distributed computer system, comprising:

means for authenticating a user, said user having a userid;

means for retrieving encrypted stored data from a first storage media, said encrypted stored data being an encryption of said stored data;

means for maintaining a database on a second storage media for correlating said encrypted stored data to a data encryption key and said userid;

means for validating an authenticated user as one of the userids listed in said database as permitted to access said encrypted stored data;

means for retrieving said data encryption key corresponding to said encrypted stored data from said second storage media; and

means for decrypting said encrypted stored data using said retrieved encryption key.

* * * * *

US006738905B1

(12) **United States Patent**
Kravitz et al.

(10) **Patent No.: US 6,738,905 B1**
(45) **Date of Patent: May 18, 2004**

(54) **CONDITIONAL ACCESS VIA SECURE LOGGING WITH SIMPLIFIED KEY MANAGEMENT**

(75) Inventors: **David W. Kravitz**, Fairfax, VA (US); **David M. Goldschlag**, Silver Spring, MD (US)

(73) Assignee: **Digital Video Express, L.P.**, Herndon, VA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/291,052**

(22) Filed: **Apr. 14, 1999**

Related U.S. Application Data

(60) Provisional application No. 60/081,739, filed on Apr. 15, 1998.

(51) **Int. Cl.**[7] **H04L 9/00**; G06F 11/30

(52) **U.S. Cl.** **713/194**; 713/193; 380/201

(58) **Field of Search** 713/193–194, 713/182; 380/201–202, 270, 278; 705/51, 52, 57

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,528,589 A		7/1985	Block et al.	
4,768,087 A		8/1988	Taub et al.	
4,796,181 A	*	1/1989	Wiedemer	705/52
4,827,508 A		5/1989	Shear	
4,905,280 A	*	2/1990	Wiedemer	463/40
4,907,273 A	*	3/1990	Wiedemer	380/230
4,908,834 A	*	3/1990	Wiedemer	380/228
4,937,866 A		6/1990	Crowther et al.	
4,945,563 A		7/1990	Horton et al.	
4,947,428 A	*	8/1990	Guillou et al.	380/240
4,980,912 A	*	12/1990	Welmer	380/232
5,010,571 A		4/1991	Katznelson	
5,029,207 A		7/1991	Gammie	
5,081,680 A		1/1992	Bennett	

(List continued on next page.)

OTHER PUBLICATIONS

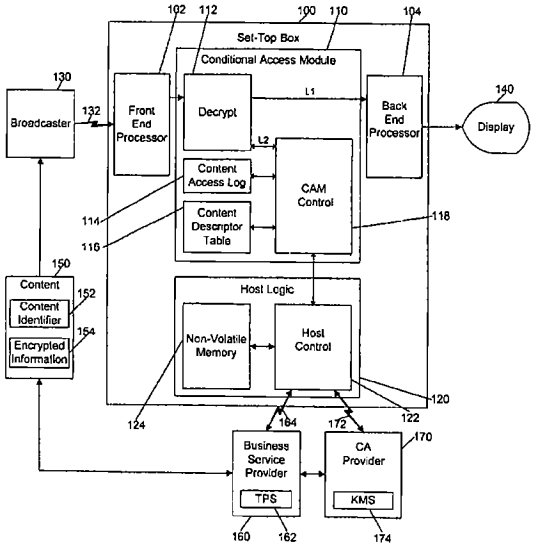
Proposal for a Conditional Access System for Terrestrial Broadcast, Canal Plus/SECA, Aug. 20, 1998.

Primary Examiner—Thomas R. Peeso
Assistant Examiner—Kambiz Zand
(74) *Attorney, Agent, or Firm*—McDermott, Will & Emery

(57) **ABSTRACT**

A method and apparatus for distributing content data from a content provider to a subscriber. The method includes encrypting content data by the content provider and providing the content data from the content provider to a broadcaster. The content provider also provides a content descriptor, including keys to decrypt the encrypted content, to a conditional access provider. The broadcaster distributes the encrypted content information to a subscriber. A business service provider negotiates with the subscriber to deliver individual content programs or packages of content programs to the subscriber for a fee. The conditional access provider distributes a content descriptor including keys necessary to decrypt the programs the subscriber selected from the business service provider. A CAM retained by the subscriber maintains a log of programs accessed, and uploads the log to the business service provider, which is used to determine the appropriate fee which the subscriber should be charged. Alternatively, the subscriber may purchase a package plan which does not require logging, and thus allows unidirectional communication.

26 Claims, 9 Drawing Sheets



U.S. PATENT DOCUMENTS

5,144,664	A	*	9/1992	Esserman et al. 380/228
5,289,271	A		2/1994	Watson
5,388,211	A		2/1995	Hornbuckle
5,499,298	A	*	3/1996	narasimhalu et al. 705/54
5,509,070	A	*	4/1996	Schull 705/54
5,671,276	A		9/1997	Eyer et al.
5,701,152	A		12/1997	Chen
5,703,951	A		12/1997	Dolphin
5,740,246	A		4/1998	Saito
5,825,883	A		10/1998	Archibald et al.
5,991,399	A	*	11/1999	Graunke et al. 380/279
6,069,647	A	*	5/2000	Sullivan et al. 725/29
6,073,122	A	*	6/2000	Wool 380/201
6,081,600	A	*	6/2000	Blanchard et al. 380/255
6,100,916	A	*	8/2000	August et al. 380/202
6,141,754	A	*	10/2000	Choy 713/200
6,182,218	B1	*	1/2001	Saito 713/176
6,249,873	B1	*	6/2001	Richard et al. 713/200
6,282,293	B1	*	8/2001	Itoh et al. 380/201
6,314,409	B2	*	11/2001	Schneck et al. 713/182

* cited by examiner

FIG. 1

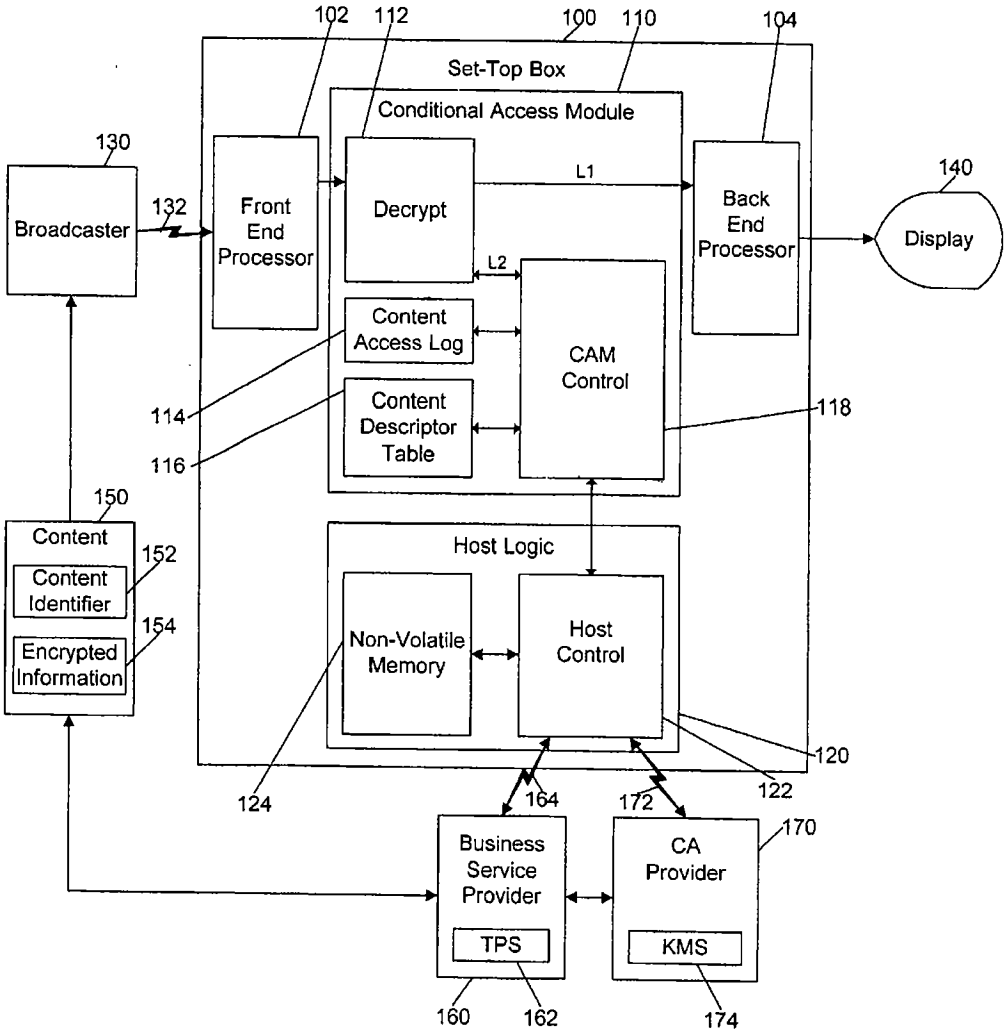

102
112
100
110
104
Set-Top Box
Conditional Access Module
130
Broadcaster
132
Front End Processor
Decrypt
L1
Back End Processor
140
Display
L2
Content Access Log
CAM Control
114
Content Descriptor Table
116
118
150
Content
Content Identifier
152
Encrypted Information
154
Host Logic
Non-Volatile Memory
Host Control
120
122
124
164
172
170
Business Service Provider
TPS
CA Provider
KMS
160
162
174

FIG. 2

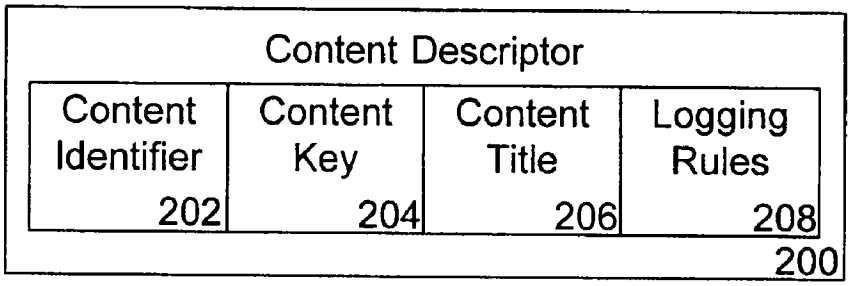

Content Descriptor
Content Identifier
202
Content Key
204
Content Title
206
Logging Rules
208
200

FIG. 3

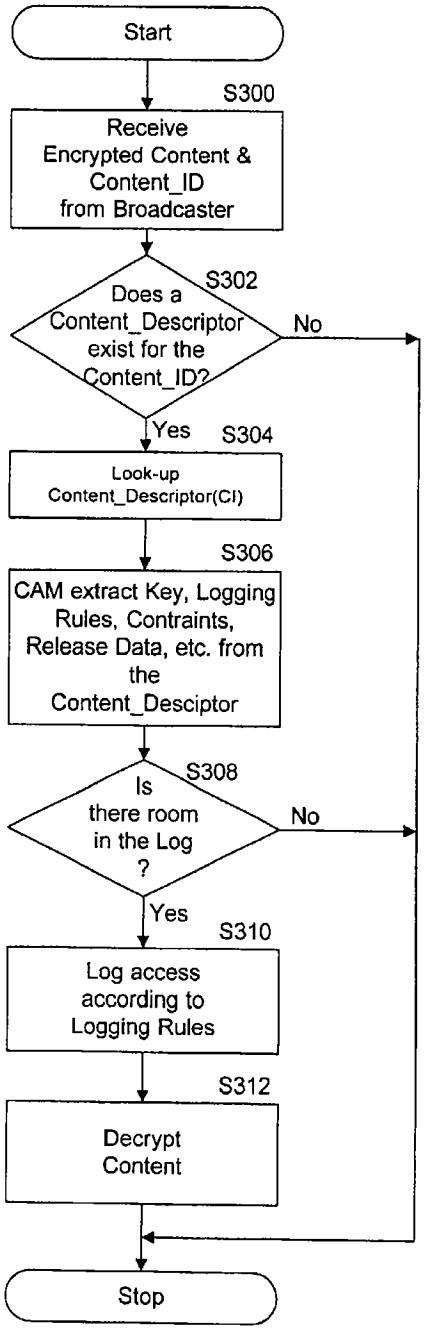

Start
S300
Receive Encrypted Content & Content_ID from Broadcaster
S302
Does a Content_Descriptor exist for the Content_ID?
No
Yes
S304
Look-up Content_Descriptor(CI)
S306
CAM extract Key, Logging Rules, Contraints, Release Data, etc. from the Content_Desciptor
S308
Is there room in the Log ?
No
Yes
S310
Log access according to Logging Rules
S312
Decrypt Content
Stop

FIG. 4

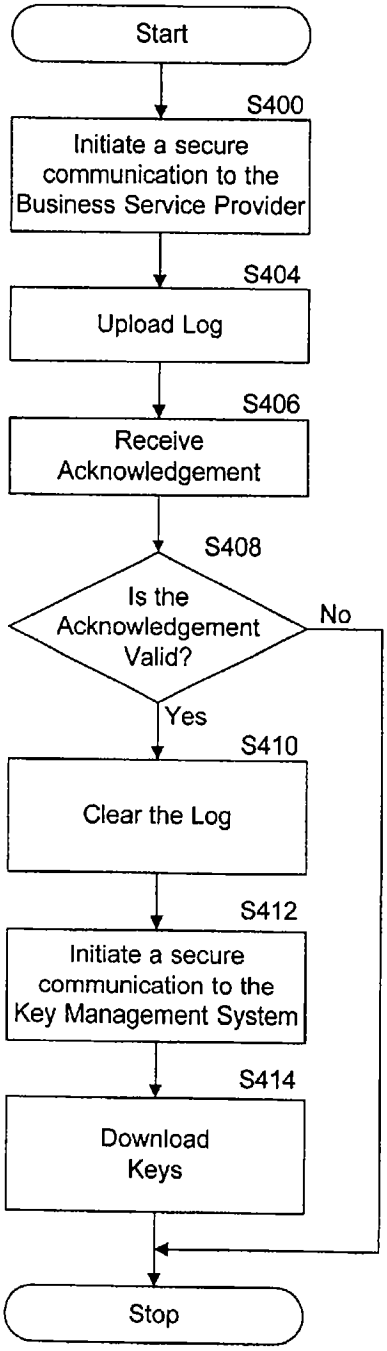

Start
S400
Initiate a secure communication to the Business Service Provider
S404
Upload Log
S406
Receive Acknowledgement
S408
Is the Acknowledgement Valid?
No
Yes
S410
Clear the Log
S412
Initiate a secure communication to the Key Management System
S414
Download Keys
Stop

FIG. 5

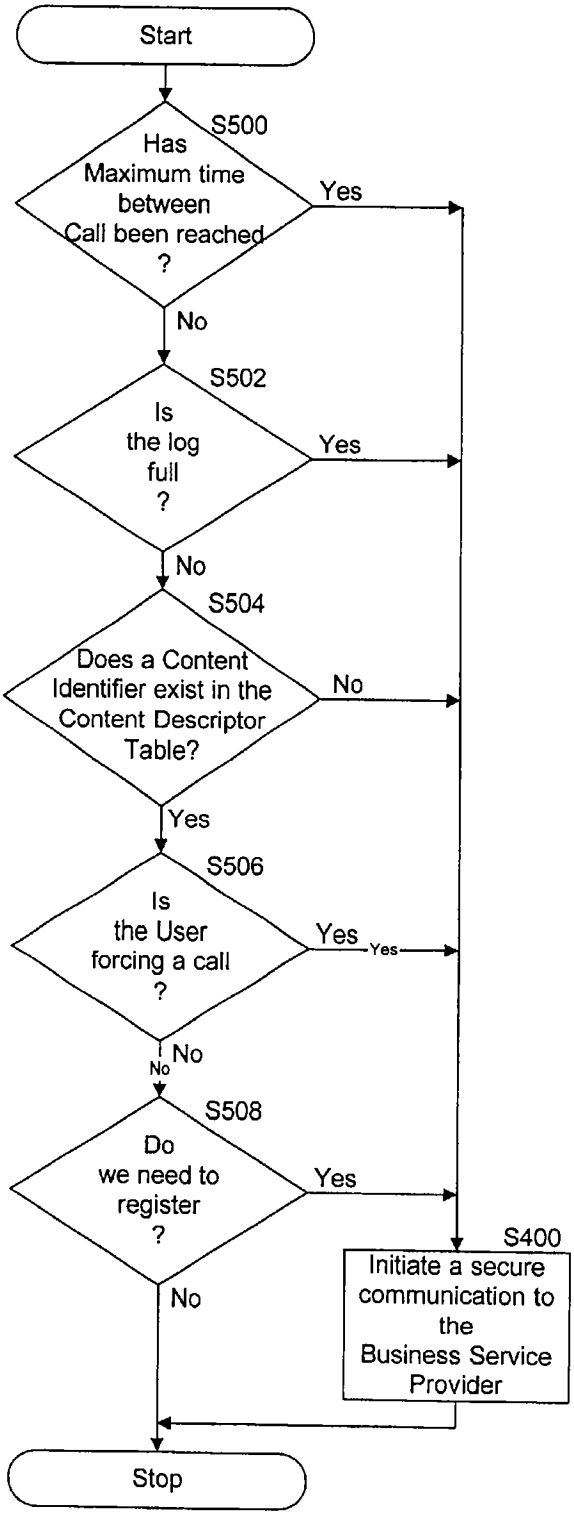

Start
S500
Has Maximum time between Call been reached ?
Yes
No
S502
Is the log full ?
Yes
No
S504
Does a Content Identifier exist in the Content Descriptor Table?
No
Yes
S506
Is the User forcing a call ?
Yes
Yes
No
No
S508
Do we need to register ?
Yes
No
S400
Initiate a secure communication to the Business Service Provider
Stop

FIG. 6

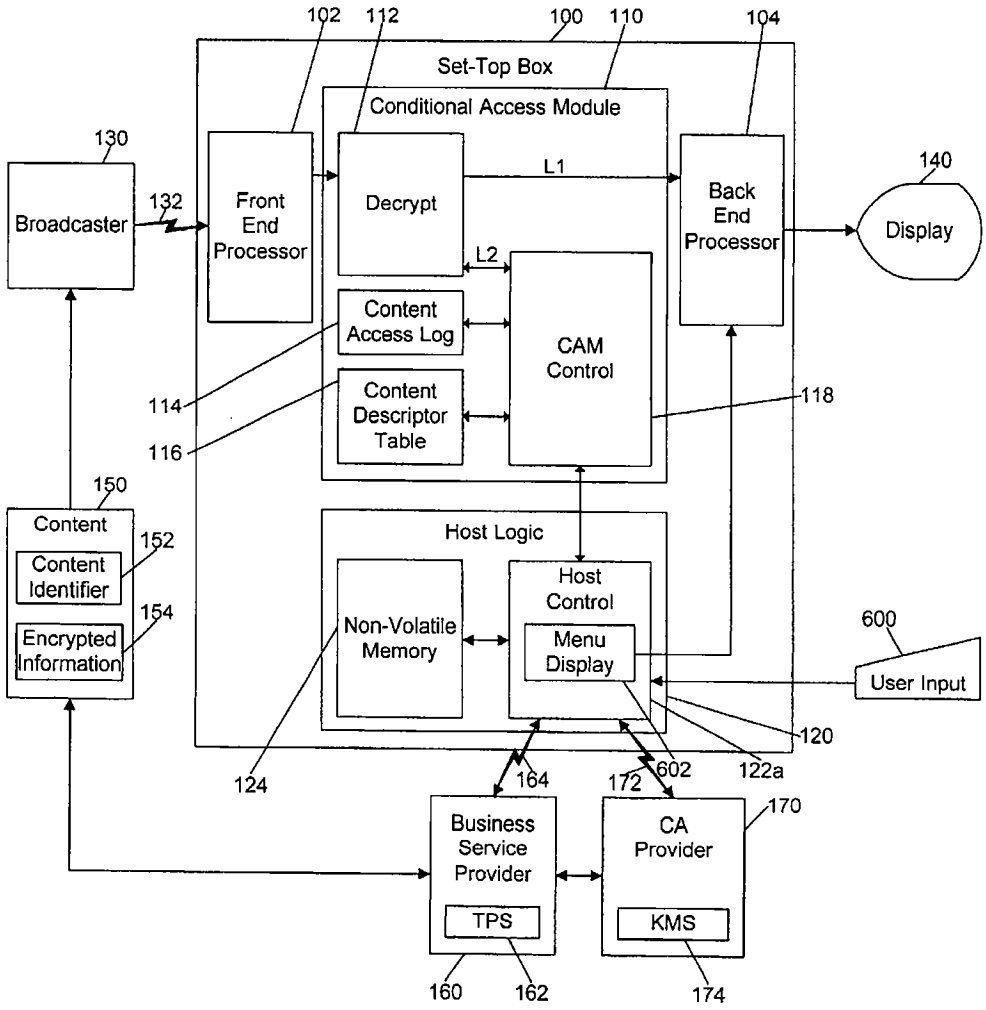
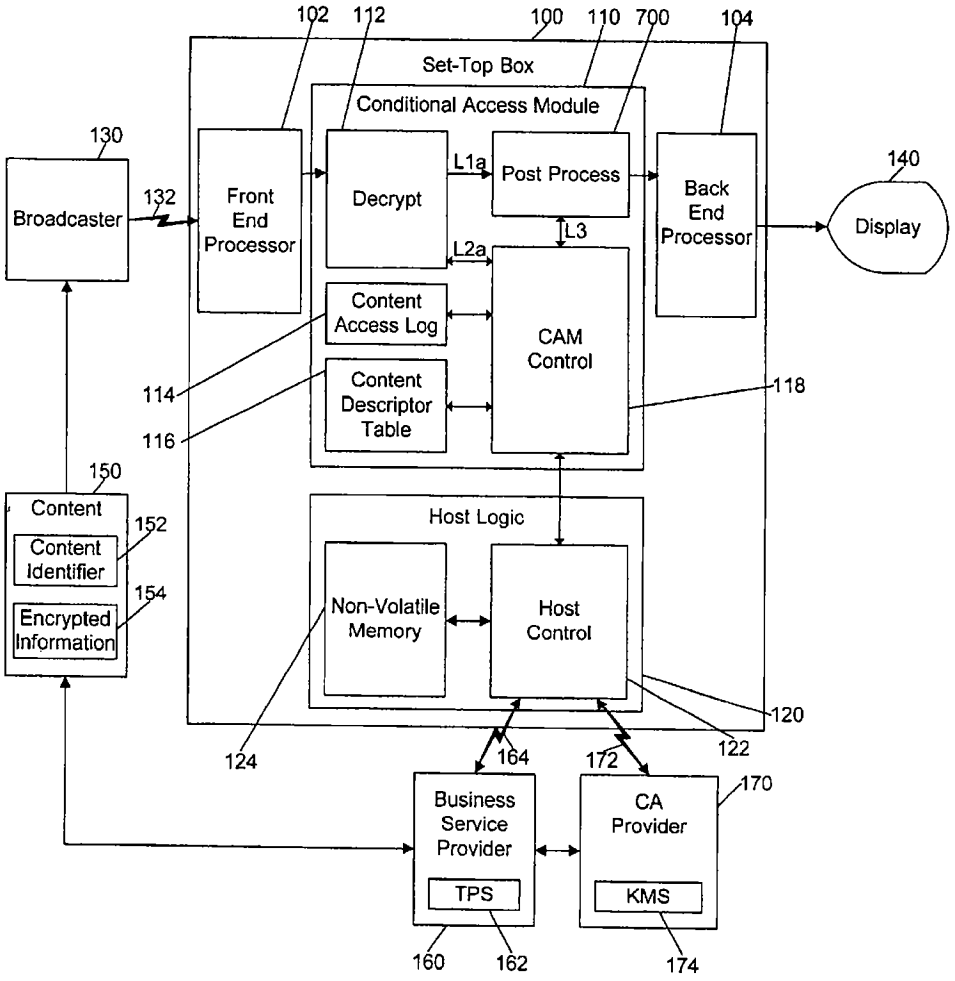

102
112
100
110
104
Set-Top Box
Conditional Access Module
130
Broadcaster
132
Front End Processor
Decrypt
L1
Back End Processor
140
Display
L2
Content Access Log
CAM Control
114
Content Descriptor Table
116
118
150
Content
152
Content Identifier
154
Encrypted Information
Host Logic
Non-Volatile Memory
Host Control
Menu Display
600
User Input
120
124
164
172
602
122a
Business Service Provider
TPS
CA Provider
170
KMS
160
162
174

FIG. 7


102
112
100
110
700
104
Set-Top Box
Conditional Access Module
130
Broadcaster
132
Front End Processor
Decrypt
L1a
Post Process
L3
L2a
Back End Processor
140
Display
Content Access Log
CAM Control
114
Content Descriptor Table
116
118
150
Content
Content Identifier
152
Encrypted Information
154
Host Logic
Non-Volatile Memory
Host Control
120
122
124
164
172
170
Business Service Provider
TPS
CA Provider
KMS
160
162
174

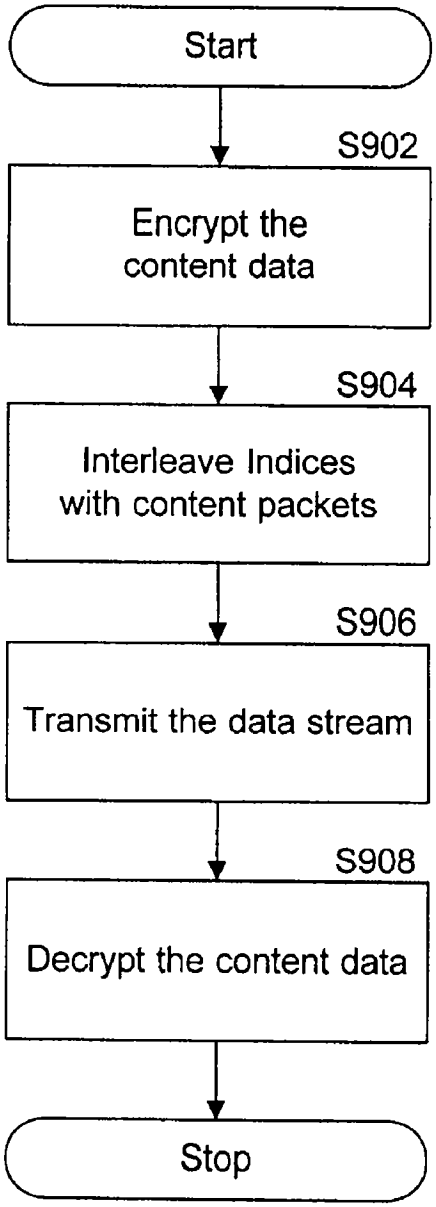

ECM1,1()
ECM1,2()
ECM1,3()
e1(p1)
e1(p2)
e1(p3)
ECM2,1()
ECM2,2()
ECM2,3()
e2(p4)
e2(p5)
e2(p6)

FIG. 8A


Index1
e1(p1)
e1(p2)
e1(p3)
Index2
e2(p4)
e2(p5)
e2(p6)

FIG. 8B

Fig. 9


Start
S902
Encrypt the content data
S904
Interleave Indices with content packets
S906
Transmit the data stream
S908
Decrypt the content data
Stop

CONDITIONAL ACCESS VIA SECURE LOGGING WITH SIMPLIFIED KEY MANAGEMENT

This application claims benefit of Provisional application Serial No. 60/081,739 filed Apr. 15, 1998.

FIELD OF THE INVENTION

The present invention pertains to a method and apparatus for allowing a subscriber to access information from several information providers, usually for a fee. More particularly, the present invention pertains to a method and apparatus for allowing a subscriber to obtain access to individual programs where the subscriber is only charged for the programs which have been accessed. The present invention also pertains to a method and apparatus for allowing a subscriber to obtain access to a package of programs in a uni-directional communication with the program service provider.

BACKGROUND OF THE INVENTION

Some types of information broadcast systems are commonly known. One type is an over the air broadcast by use of UHF and VHF signals which can be freely obtained by anyone who can pick up such signals with an appropriate antenna. These systems generate revenue by selling airtime for commercial broadcasts. A second type of system is subscription system where a subscriber subscribes to the broadcast system which is provided, for example, through a cable line or a satellite dish.

The subscription systems protect the content of the information in the broadcast from being read by non subscribing cable or satellite dish users by complex encryption schemes. Such schemes often use a new encryption method for every discrete period of time within the broadcast, i.e. every 3 seconds. These schemes also commonly require the subscriber to have a conditional access module (CAM) which receives and decrypts the encrypted content data. In order for the CAM to decrypt the information, ECMs (Entitlement Control Messages) are interleaved with data packets containing the content information which instruct the CAM how to decrypt the content information. However, in typical subscription broadcast systems, the subscriber is only able to subscribe to one information provider, i.e. they are single subscription systems. Generally, the CAM of one information provider will not be able to interpret the ECMs of another information provider. This limits the subscriber's ability to freely choose programs since the subscriber is only able to receive the programs offered by one particular information provider.

One solution is for the subscriber to subscribe to several different information providers. This would require the subscriber to have a CAM for each provider and to be charged separately by each provider. Additionally, a subscriber may have access to several information providers in his/her geographic region, such as, several local broadcast service providers and cable service providers. Clearly, this is an expensive and highly undesirable choice.

Another solution is known as SIMULCRYPT, in which the ECMs of a plurality of information providers are interleaved throughout the content of the broadcast. This allows different service providers to use the same content information, thus sharing their resources. However, SIMULCRYPT does not provide the subscribers any more flexibility since the CAM of the subscriber can only interpret one of the ECMs, i.e. only the ECM of their information provider. Another problem of SIMULCRYPT is that it

involves a high degree of cooperation between competing broadcasters, such as requiring competing broadcasters to provide confidential ECM information to be broadcast on the broadcast mechanisms.

Furthermore, the advent of HDTV (high definition TV) will likely enable traditional air broadcast information providers to provide subscription air broadcasts, since digital data in the broadcast could easily be encrypted using known techniques. A single broadcaster is expected to be able to allocate their HDTV broadcast channel (provided to the broadcasters by the Federal Communications Commission (FCC)) into as many as four, or more, air broadcast channels. Additionally, many of those channels may be encrypted for subscription channels, i.e. up to three of the HDTV channels. Since there are often at least four television networks which broadcast in single geographic area, the number of subscription information providers available to a single subscriber will greatly increase if each of the networks provides a plurality of subscription channels. This is even a greater problem for a subscriber who lives on a border between two geographic broadcast areas, such as between Baltimore and Washington D.C., and thus would potentially have to deal with twice as many subscription channels from twice as many information providers. It would be very inconvenient for a subscriber to subscribe to each of these subscription channels since the subscriber would need to have several different CAMs to decrypt the several different channels, and the subscriber would have to pay each of the several. different information providers.

One possible solution is for the networks to closely cooperate with each other, trading ECMs and encryption equipment. Thus, the CAM provided by one network, for example ABC, would be able to decrypt the content information provided by other networks, and ABC could collect the subscription fee from the subscriber and redistribute the appropriate amounts to the other networks. However, such a solution appears to require a very high degree of cooperation and trust, neither of which are likely to exist in the competitive broadcast industry. For example, if the security of another network, such as NBC, was compromised, ABC would have little incentive to repair any ABC CAMs to correct the security breach affecting NBC. Additionally, the degree of cooperation would multiply to provide subscribers on a border of adjacent broadcast areas with the same service.

A proposed solution for obtaining access to encrypted broadcast data is disclosed in U.S. Pat. No. 5,010,571 to Katznelson. Katznelson disclose a process for obtaining access to encrypted information stored on a disc by using an authorization and key distribution terminal (service provider) to send the decryption keys to the customer, after the customer has requested such keys in a customer initiated communication. However, in this proposed solution, the customer must contact the service provider prior to accessing any discs (information), thus offline operations and/or a unidirectional operations are not possible. Additionally, it may be a great inconvenience to the customer to request access every time it is desired, since such request would take additional time and effort by the customer, both of which may be significant depending on the speed and availability of connection to the service provider.

Another approach to obtaining access to encrypted broadcasted information is proposed in U.S. Pat. No. 5,703,951 to Dolphin. Dolphin discloses a procedure wherein a disc (CD-ROM) containing a plurality of encrypted magazines is distributed to customers, who are able to access the individual magazines to which they have a corresponding key

stored on a PCMCIA card. If the customer does not have the corresponding key on their PCMCIA card, then they can contact a service provider for such key. However, this proposal is per publisher based; that is, there is no procedure for providing the customer access to magazines published by several different publishers. In fact, the only way that the user could get magazines from other publishers is to have competing publishers share confidential encryption information, and share billing receipts with each other. This is an undesirable business procedure, since a publisher will not be able to truly compete with its competitors. Another related drawback of this proposal is that the customer has very limited flexibility in choosing which magazines to access, since the customer is limited to the particular magazines which happen to be stored on the CD-ROM distributed by the publisher.

SUMMARY AND OBJECTS OF THE INVENTION

It is an object of the invention to provide subscribers with a cost effective method for providing a plurality of subscription services in a geographic area.

It is another object of the invention to provide subscribers at the borders of adjacent broadcast areas with a cost effective method of obtaining plurality of subscription services.

It is yet another object of the invention to allow a user to choose to receive a package of content from a service provider, or to receive individual programs of content from the service provider.

It is yet another object of the invention to allow a subscriber to receive content data through an off-line operation.

It is yet another object of the invention to allow a subscriber to receive content data through a unidirectional communication with a service provider.

To achieve the foregoing and other objects and in accordance with the purpose of the present invention, as embodied and broadly described herein, the method of this invention may comprise a method of distributing content data from a content provider device to a subscriber, the content provider including an encryption device, the subscriber having a set top box including a decryption device, the method comprising the steps of: encrypting content using the encryption device; distributing the encrypted content with content identifiers from the content provider to the subscriber; distributing keys for decrypting the encrypted content from the content provider to a service provider; distributing the keys from the service provider to the subscriber; and accessing the content by decrypting the content using the decryption device.

In a further aspect of the present invention, in accordance with its objects and purposes, the method hereof may also preferably comprise the steps of: logging access of the content by the subscriber in a log contained on a non-volatile memory; and sending the log to the service provider.

In yet a further aspect of the present invention, in accordance with its objects and purposes, the method hereof may further preferably include distributing rules with the keys, wherein the steps of accessing the content and logging access of the content are performed according to the rules.

In yet a further aspect of the present invention, the step of distributing the keys may further preferably include distributing the keys by broadcasting the keys to the subscriber.

In yet a further aspect of the present invention, the step of distributing the encrypted content may further preferably be performed by multiple content providers, and the step of distributing keys is performed by multiple service providers, and wherein the subscriber interacts with at least one of said multiple content providers and at least one of said multiple service providers.

In yet a further aspect of the present invention, the step of distributing the keys may further preferably include distributing encrypted content from a single content provider to a plurality of broadcasters; and broadcasting the encrypted content to the subscriber.

In yet a further aspect of the present invention, wherein the set top box may further preferably include a display device and a menu navigation device, and may further preferably comprise the steps of: displaying a menu on said display device; receiving input from said menu navigation device; and controlling access to content based on the received input.

Another method according to the present invention may comprise a method of distributing content data from a content provider to a subscriber, the subscriber having a set top box including a decryption device, the method comprising the steps of: encrypting the content data using a content key and an index, in a series of indices, corresponding to a content packet, in a series of content packets; forming a data stream by interleaving the series of indices with a series of content packets; transmitting the data stream to the set top box; and decrypting the content packet using a content key and the index corresponding to the content packet in a series of indices.

In a further aspect of the present invention, in accordance with its objects and purposes, the other method hereof may also preferably comprise encrypting and decrypting by hashing the indices with the content key.

An apparatus according to the present invention may comprise an apparatus for receiving and decrypting encrypted content data from a content provider to a subscriber via a broadcaster, using a content descriptor provided by a service provider, the apparatus comprising: a receiver for receiving the encrypted content; a decryptor for decrypting the encrypted content using the content descriptor; and a controller for controlling the decryptor and for communicating with the service provider, thereby providing the subscriber access to the content data.

In a further aspect of the present invention, in accordance with its objects and purposes, the apparatus hereof may also preferably comprise, a logging device for logging access to the content data, wherein the controller controls the logging device log access to the content data according to rules contained in the content descriptor.

In yet a further aspect of the present invention, in accordance with its objects and purposes, the apparatus hereof may also preferably comprise communication control logic, the communication control logic being responsive to the controller to communicate with the service provider.

In a further aspect of the present invention, in accordance with its objects and purposes, the apparatus hereof may also preferably comprise a menu generator for generating a menu, having a plurality of operations, to be displayed on a display device; and a menu navigator for providing a user selection of said operations to said controller.

The invention is advantageous over the SIMULCRYPT because it provides the conditional access control to a subscriber through a different channel than the content, thus broadcaster can maximize its business by disseminating the broadcast to as many people as possible without investing resources to protect the content of the broadcast.

Furthermore, the broadcaster is no longer required to allocate space on the broadcast transmission for transmitting conditional access control data such as keys to decrypt the content data. Additionally, unlike SIMULCRYPT, the present invention does not require significant cooperation between competing broadcasters or competing content providers, and thus does not require them to share confidential conditional access information or subscription fees.

Additional objects, advantages and novel features of the invention will be set forth in part in the description which follows, and in part will become apparent to those skilled in the art upon examination of the following or may be learned by practice of the invention. The objects and advantages of the invention may be realized and attained by means of the instrumentalities and combinations particularly pointed out in the appended claims.

BRIEF DESCRIPTION OF THE DRAWINGS

The accompanying drawings, which are incorporated in and form a part of the specification, illustrate several embodiments of the present invention and, together with the description, serve to explain the principles of the invention. In the drawings:

FIG. **1**, illustrates a schematic diagram of the present invention.

FIG. **2**, illustrates the content of the content descriptor.

FIG. **3**, illustrates a logical flow of the operation of the invention.

FIG. **4**, illustrates a logical flow for communicating with the business service provider and the CA provider.

FIG. **5**, illustrates a conditions for triggering a communication.

FIG. **6**, illustrates another embodiment of the schematic diagram of the present invention.

FIG. **7**, illustrates yet another embodiment of the schematic diagram of the present invention.

FIGS. **8**A and **8**B, illustrate yet another embodiment of the invention which uses indices to distribute the content information.

FIG. **9**, illustrates a logical flow of the embodiment shown in FIG. **8**B.

DETAILED DESCRIPTION OF THE INVENTION

Reference will now be made in detail to the present preferred embodiments of the invention, examples of which are illustrated in the accompanying drawings.

FIG. **1** illustrates a first embodiment of the invention. This embodiment of the invention comprises five basic components which interact with each other. The content provider **150** generates content, such as movies, adds content identifier (CI) data using a content identifier device **152**, and encrypts the content information using an encryption device **154**. The content provider **150** also generates keys to decrypt the encrypted content. A broadcaster **130** broadcasts the encrypted content with the CI data to the subscriber, who receives the content and decrypts it via a set top box **100**. The content provider **150** also provides a business service provider **160** with appropriate keys to decrypt of a plurality of programs generated by the content provider **150**. The business service provider **160** interacts with the subscriber to determine what programs the subscriber would like to access, i.e. subscribe to, and provides the corresponding keys to a conditional access (CA) provider **170**. The CA provider **170** installs, or otherwise provides, a conditional access module (CAM) **110** in (or for) the set top box **100** of the subscriber, and provides keys to the subscriber which allows the subscriber to decrypt specific programs of content data. The CAM **110** receives the keys from the CA provider **170**, and allows the set top box **100** to obtain access to receive content (i.e. decrypt the content) whose CI data corresponds to the keys received from the CA provider **170**.

The set top box **100** is typically located in the home of the subscriber and is used in conjunction with a display device **140**, e.g. a television set. The set top box **100** includes a front end processor **102** which receives broadcast information from a broadcaster **130** by conventional receiving techniques. The information may be transmitted from the broadcaster **130** by conventional air broadcast transmission, by cable transmission, by satellite transmission, by supply of recorded media (such as ROMs, disks or tapes), by internet transmission, or by any other type of transmission. The front end processor **102** sends the received signal to a CAM **110** which decrypts the content data contained in the signal using a decryption device **112**, and forwards the decrypted content data to a backend processor **104** over signal line L1. The backend processor **104** converts the decrypted content data into a form which may be displayed on the display device **140**. Preferably, the back end processor **104** is a MPEG processor.

The decryption device **112** may be a conventional decryption device as readily understood by those of ordinary skill in the art. The decryption device **112** receives the broadcast signal containing the content data and the CI data, and forwards the CI data to a CAM control device **118** over a bidirectional signal line L2. The CAM control device **118** accesses a content descriptor table **116** to determine if the subscriber is authorized to decrypt the content data. Specifically, the content descriptor table **116** contains a list of content descriptors (CDs) which contains unique information for particular program of content which can be used by the CAM control device **118** to determine the decryption key, the title, and any rules governing access to and logging of the content. The presence of the CD in the content descriptor table **118** is positive authorization implicitly granting the subscriber access to the content. This positive authorization may be constrained by explicit negative authorizations in the rule: for example, time windows during which the content may be accessed. The CAM control device **118** provides such authorization or negative authorization to the decryption device **112** over the bi-directional signal line L2. When authorization is given from the CAM control device **118** to decrypt the content data, a content access log **114** logs the access to the content. The content access log **114** is preferably a non-volatile memory.

A host logic **120** controls the communication between the CAM control device **118** and the business service provider **160**, and between the CAM control device **118** and the CA provider **170**. Specifically, the host logic **120** contains host control logic **122** which controls a communication device (not shown), such as a modem, a DSL, or a parallel link, through which the CAM control device **118** interacts with the business service provider **160** to allow the subscriber to select which programs the subscriber desires to access via subscription. The business service provider **160** contains a transaction processing system (TPS) **162**, which preferably uploads the log of accessed content stored in the. content access log **114**. An example of a TPS system which may be used as TPS **162** in conjunction with the present invention is described in commonly assigned U.S. patent application Ser. No. 09/092,177 to Oren et al., filed on Jul. 24, 1998. The

business service provider **160** may use the log of accessed content to bill the subscriber, to determine the programs, or type of programs desired by their subscribers, etc. The business service provider may also use the log to determine how the subscription fee, paid by the subscriber, should be distributed among a plurality of content providers in the form of royalties. The business service provider **160** also provides the selected program information to the CA provider **170**.

The CA provider **170** contains a key management system **174** which contains a plurality of keys associated with the plurality of programs offered by the content provider **150**. The CA provider **170** securely provides the appropriate keys to the CAM control device **118**, which in turn provides the keys to the decryption device **112** to decrypt the content of the desired program. The CA provider **170** also preferably provides new keys to the host control device **122** when the content provider **150** changes the keys. Typically the are supplied to the CA provider **170** by the content provider **150** prior to such further distribution.

The host logic also contains a non-volatile memory **124** which, may for example contain billing information, such as a particular billing plan; communication information which, may for example instruct the host control **122** to initiate a communication with either the business service provider **160** or the CA provider **170**; or other information as will be discussed later.

The CI data is unique to each program of content which particularly identifies the particular program of content. The content identifier may be as simple as an index into a table of content descriptors (CDs). As shown in FIG. **2**, a CD **200** preferably contains a content identifier (CI) **202**, a content key **204**, content title **206**, and logging rules **208**. The logging rules **208** may include such information as a release date, which prevents access to the content data until such release date, or a particular billing method which should be used for the accessed content. The logging rules **208** may be established by the content provider **150** or the business service provider **160**.

The CAM **110** preferably securely logs access to content rather than charging for the right to access the content. The amount of information contained in a logging operation may be controlled by parameters in the CAM control device **118** and in the CD logging rules **208**. The parameters may include the number of bytes of content received, the time of receiving the content, or a particular usage code contained in the CD. Additionally, content logging may involve two stages: an initial entry which logs the overall access to the content (e.g., who is accessing what), and subsequent entries which further describe the access to the content (e.g., what part of the content is accessed, how much, and when).

The CAM control device **118** securely and periodically sends the secure log to the CA provider **170**, who in turn may pass it along to the business service provider **160**. Alternatively, the CAM control device **118** may securely and periodically send the secure log directly to the business service provider **160**. The content access log **114** preferably can only be cleared after the CAM control device **118** receives a positive acknowledgment that the log has been successfully transferred.

The embodiment is FIG. **1** has been described above as containing five basic components which interact with each other. This was done to illustrate the operation of the invention, and is not intended to constraint the specific structure of the invention. It will be clear to the ordinary artisan that several of the components may be combined into one entity or considered as further broken down into discrete entities. For example, the business service provider **160** and the CA provider **170** may be joined as a single entity called a service provider. Additionally, the content provider **150** may be joined with the broadcaster **130** as a single entity. Different aspects of the set top box **100** may be further broken down into and considered as discrete components. Any other combinations that do not significantly alter the operation of the invention will be readily apparent to those of skill in the art to be encompassed by the present invention.

Additionally, the embodiment described in FIG. **1** illustrates one broadcaster **130** and one content provider **150**. This was done to simplify the illustration of the operation of the invention, and is not intended to place constraints on the specific structure of the invention. Particularly, a single content provider **150** may distribute content to a plurality of broadcasters, distribute conditional access information to a plurality of CA providers and receive royalties from a plurality of business service providers, which may be considered to be collectively represented by broadcaster **130**, CA provider **170**, and business service provider **160** respectively. Furthermore, a single broadcaster may receive and broadcast content from a plurality of content providers, which may be considered to be collectively represented by content provider **150**. Additionally, a single business service provider and/or CA provider may receive conditional access information from a plurality of content providers, which may be considered to be collectively represented by content provider **150**. Additional component combinations and separations will suggest themselves to the artisan.

The operation of the CAM **110** is illustrated in FIG. **3**. As shown in step **S300**, the CAM **110** receives encrypted content and CI data from the broadcaster **130** through the front end processor **102**. The CAM control device **118** accesses the content descriptor table **116** to determine if a CD exists for the CI data (**S302**). If a CD does not exist for the CI data, then access is denied and the process stops. If a CD does exist for the CI data, then the CAM control device **118** accesses the CD corresponding to the CI data (**S304**) and extracts the key and any logging rules that may exist (**S306**). Next, the CAM control device **118** determines if there is enough memory space in the content access log device **114** to log the access of the content information (**S308**). If there is not enough space, then access is denied and the process stops. If there is enough space, then the access of the content is logged if the logging rules permit access (**S310**), and the content is decrypted (**S312**) by the decryption device **112**.

For security purposes, the set top box **100** preferably periodically (or non-periodically but from time to time) contacts the business service provider **160** and the CA provider **170** or is contacted by those providers. where the set top box initiates communication, this communication preferably follows the protocol illustrated in FIG. **4**. The communication may be initiated after any of the determinations made in FIG. **5**. Specifically, a communication is initiated: if a maximum time period has elapsed since a prior communication (**S500**); if the content access log **114** no longer has enough available free memory space to log access to content data (**S502**); if a CI is not contained in the content descriptor table **116** (**S504**); if the subscriber requests a communication (**S506**); or if the set top box needs to register with the business service provider **160** or the CA provider **170**, as in a newly installed set top box (**S508**). As shown in step **S400**, the CAM control device **118** initiates a secure communication to the business service provider **160**.

Once a secure communication is initiated, the CAM control device **118** uploads the log contained in the content

access log **114** to the business service provider **160** (S404). The CAM control device **118** then preferably receives an acknowledgment signal from the business service provider **160** (S406). The CAM control device **118** determines if the acknowledgment is valid (S408). If the acknowledgment is not valid, the protocol stops. If the acknowledgment is valid, the CAM control device **118** clears the content access log **114** (S410) to allow additional content which is accessed to be logged. After clearing the content access log **114**, the CAM control device **118** initiates a secure communication with the CA provider **170** (S412), and downloads CDs, including keys required to decrypt the content data (S414). In this manner, the CAM control device **118** is able to learn new CDs to access new content, some of the CDs may only be accessible by contacting the CA provider.

The untrusted CAM control device **118** may also use the CI data in various way, such as to index into its own tables to determine the fee schedule for content. In that case, the CAM control device **118** runs a simplified billing system which can attempt to predict how much the subscriber is charged. But the results are guidance and not a guarantee since the true billing system preferably has global knowledge that may not be present locally. For example, the CAM control device **118** may not be aware that the subscriber has previously purchased unlimited rights to the content data from the business service provider **160**.

The present invention lends itself to a market based relationship between the participants in broadcast transmission. In one scenario, the content provider **150** owns the content and the keys required to decrypt the content. In this scenario, the content provider **150** contracts with broadcasters to widely distribute the content, while conservatively using the broadcast spectrum, and with CA providers to securely distribute the keys. The content provider **150** may freely distribute the keys to the CA provider **170**, or CA providers may pay a flat fee for these keys (based on their number of subscribers, for example), or may return to the content provider **150** a portion of the transaction revenue they charge subscribers for using those keys. Additionally, the keys themselves may be sold, leased or traded among several different CA providers, content providers, business service providers, broadcasters, or even subscribers through their subscription plans. In any case, the subscribers preferably only pay for access to the content or for a particular subscription plan, and not for access to the keys themselves.

The content providers preferably receive fees from CA providers based on the subscriber's behavior, such as by the particular content which is accessed. This revenue may be based directly on the subscriber's use of the content keys, or based on subscription packages that entitle the subscriber access to certain content packages. Furthermore, the business service provider **160** may bundle individual programs as packages so that different subscribers may purchase the packages. Alternatively, the business service provider **160** may allow individual subscribers to select individual programs which may be contained in a package. Accordingly, different subscribers may purchase the same program in different ways. Each CA provider **170** may also be penalized for compromised security, for example the number of counterfeit CA modules found to be used or sold based on a particular CA provider's design. This penalty provides incentive to the CA provider **170** to increase security, and also give new competitors to the market mechanism and incentive to lower their costs, by providing a more secure system. Properly balanced incentives will guide the market to the appropriate tradeoffs between security costs, fraud, revenue, and profit.

A principle advantage of the present invention is that CD distribution is separated from the content distribution. The CA provider **170** decides on the most efficient way to deliver CDs to the subscribers. For example, the CA provider **170** may purchase bandwidth from broadcasters and broadcast the CDs to their customers, or use a paging network. Alternatively, the CA provider **170** may deliver the CDs to individual subscribers over a phone line, via a modem or other communication device. This presents the CA provider **170** with great flexibility, it can deliver CDs to subscribers living between two broadcast regions even if their primary market is only in one of those regions, when it is not cost effective to broadcast to the other customers. Furthermore, since the CDs contain the single key needed to decrypt a particular piece of content, the CA provider **170** does not need to pay for broadcasting ECMs in that region.

Another principle advantage of the present invention is that it facilitates off-line operations. That is, the subscriber does not need to contact anyone to obtain access to the content to which the CAM **110** has CDs in the content descriptor table **118**. Additionally, even if the CAM **110** does not contain the corresponding CD, the subscriber only needs to contact the CA provider **170** or the business provider **160** to obtain access, not the content provider **150** or the broadcaster **130**.

In another embodiment, an example of which is illustrated in FIG. **6**, the host control device **122***a* includes a menu display device **602**. The menu display device **602** provides a menu display to the backend processor **104** which converts it to a form which can be displayed on display device **140**. The host control device **122***a* receives subscriber input from an input device **600**, which may control selection of content to be accessed and communication with the business service provider **160** and the CA provider **170**. Preferably, the subscriber input is in the form of a selection of an item displayed in the menu generated by the menu display device **602**. The items displayed on the menu may include a listing of programs available, i.e. being broadcast, and control functions of the set top box **100**.

In another embodiment, an example of which is depicted in FIG. **7**, a post processing device **700** receives decrypted content on line L1*a* and manipulates the content data after it is decrypted by the decryption device **112** in order to reduce its value before passing it along to the subscriber, via the backend processor **104**. The post processing device **700** receives post processing information, which instructs the post processing device how to manipulate the decrypted content, from the CAM control device **118** over signal line L3. The CAM control device **118** receives CI data over bi-directional signal line L2*a*. Such manipulation may include the addition of identifying data, such as watermarks, to the decrypted content data. This makes it possible to detect if a subscriber is redistributing the decrypted content data. The post processing may also provide interface protection to protect the content between the CAM **110** and the backend processor **104**.

Another embodiment of the invention will now be described in connection with FIGS. **8**A and **8**B. FIG. **8**A illustrates a known technique of broadcasting by interleaving ECMs with packets of content data. In order for the broadcaster to support more than one CA provider, using SIMULCRYPT, a different ECM would have to be provided for each content provider. Specifically, FIG. **8**A illustrates a broadcast stream using SIMULCRYPT with three CA providers, each having their own ECMs interleaved within the content data. Furthermore, since each broadcaster interleaves a new ECM at frequency of every 3–20 seconds of

broadcast, a long program of content has many ECMs. If these ECMs are required for the content decryption key, then a content descriptor table on a set top box **100** would need to be very large to store all of the ECMs. As a result, the content descriptor table **116** in a set top box **100** (eg., FIGS. **1, 6** or **7**) would have to be very large to store the ECMs. This is too burdensome to be practical.

A solution to this problem is illustrated by FIG. **8**B. Particularly, index data is interleaved with the content packets by each content provider. Each content provider preferably uses different index data, and the index data is interleaved at a frequency of every 3–20 seconds of broadcast. However, the index data itself is used by the CAM control device **118** (eg., FIGS. **1, 6** or **7**) to determine the decryption key. Specifically, the CAM a control device **118** can compute the decryption key by using the hash of the index, i.e. decryption key=hash(index, key, index). This allows standardized decryption and encryption operations to be used by CAMs from different CA providers, since the hash computation enables each CAM control device to compute the new key from the content's key in the content descriptor and the current index. This allows a significant reduction in the memory space required on the set top box **100**, allows a single subscriber to interact with a large number of content providers easily, and does not require cooperation between broadcasters.

The operation of the embodiment of the invention illustrated in FIG. **8**B is shown in FIG. **9**. Particularly, the content data is encrypted using the hash function using a key value and an index value (**S902**), i.e. encryption key=hash(index, key, index). Then the index value is interleaved with associated content packets to form a data stream (**S904**). The data stream is then transmitted to the subscriber (**S906**). The subscriber decrypts the content data using the key value and the index value associated with the particular content packet using a hash function (**S908**), i.e. decryption key=hash (index, key, index). Furthermore, the data stream illustrated in FIG. **8**B may be transmitted via a point to point broadcast, such as used by cellular phones, in which a specified subscriber receives a specified data stream. Alternatively, it may be transmitted in a multicast broadcast in which a large number of subscribers receive the same data stream.

The foregoing description of the preferred embodiments of the invention have been presented for purposes of illustration and description. It is not intended to be exhaustive or to limit the invention to the precise form disclosed, and obviously many modifications and variations are possible in light of the above teaching. It is intended that the scope of the invention be defined by the claims appended hereto.

We claim:

1. A method of distributing content data from a content provider device to a subscriber, the content provider device including an encryption device, the subscriber having a set top box including a decryption device, the method comprising the steps of:

encrypting content using the encryption device;

distributing the encrypted content from the content provider to the subscriber;

distributing keys, for decrypting the encrypted content, from the content provider to a service provider;

distributing the keys from the service provider to the subscriber; and

accessing the content by decrypting the content using the decryption device, capable of decrypting content from several different content provider devices,

wherein the keys are distributed to the subscriber temporally independent of the distribution of the content to the subscriber.

2. The method according to claim **1**, further comprising the steps of:

logging access to the content by the subscriber in a log contained on a non-volatile memory; and

sending the log to the service provider.

3. The method according to claim **2**, wherein the step of distributing the keys further includes distributing rules with the keys, and wherein the steps of accessing the content and logging access of the content are performed according to the rules.

4. The method according to claim **2**, further comprising the steps of:

generating a subscriber bill at the service provider based on the log; and

computing royalties to be paid to the content provider.

5. The method according to claim **4**, wherein the step of generating a subscriber bill comprises allocating charges based on access to an individual program.

6. The method according to claim **4**, wherein the step of generating a subscriber bill comprises allocating charges based on access to a package of programs.

7. The method according to claim **1**, further comprising the step of:

logging access to the content by the subscriber in a log contained on a non-volatile memory **2**, wherein said step of accessing content is achieved without contacting the service provider.

8. The method according to claim **2**, wherein the service provider includes a conditional access provider for distributing the keys, and a business service provider for receiving said log.

9. The method according to claim **2**, further comprising the steps of:

receiving an acknowledgement signal from the service provider;

determining if the acknowledgement signal is valid; and

clearing the log if the acknowledgement is determined to be valid.

10. The method according to claim **2**, further comprising the steps of:

receiving an acknowledgement signal from the service provider;

determining if the acknowledgement signal is valid; and

if valid, downloading keys from the service provider.

11. The method according to claim **1**, wherein the step of distributing the keys comprises broadcasting said keys to the subscriber.

12. The method according to claim **1**, wherein said step of distributing the encrypted content is performed by multiple content providers, and the step of distributing keys is performed by multiple service providers, and wherein said subscriber interacts with at least one of said multiple content providers and at least one of said multiple service providers.

13. The method according to claim **1**, wherein the step of distributing the encrypted content further comprises the steps of:

distributing encrypted content from a single content provider to a plurality of broadcasters; and

broadcasting the encrypted content to the subscriber.

14. The method according to claim 1, wherein the set top box further includes a display device and a menu navigation device, further comprising the steps of:

displaying a menu on said display device;

receiving input from said menu navigation device; and

controlling access to content based on the received input.

15. The method according to claim 1, further comprising the step of:

processing the decrypted content to inhibit distribution of the decrypted content to more than one display device.

16. The method according to claim 1, further comprising the step of transmitting the content data through a secure point to point broadcast.

17. The method according to claim 1, further comprising the step of transmitting the content data through a multicast broadcast.

18. A method of distributing content data from a content provider to a subscriber, the subscriber having a set top box including a decryption device, the method comprising the steps of:

encrypting the content data using a content key and an index, in a series of indices, corresponding to a content packet, in a series of content packets;

forming a data stream by interleaving the series of indices with a series of content packets;

transmitting the data stream to the set top box; and

decrypting the content packets using a content key and the index corresponding to the content packet in a series of indices using the decryption device, the decryption device capable of decrypting content from several different content provider devices,

wherein the keys are distributed to the subscriber temporally independent of the distribution of the content packets to the subscriber.

19. The method according to claim 18, wherein the steps of encrypting and decrypting are performed by hashing the indices with the content key.

20. An apparatus for receiving and decrypting encrypted content data which is distributed from a content provider to a subscriber via a broadcaster, the apparatus comprising:

a receiver for receiving the encrypted content;

a decryptor for decrypting the encrypted content using a content descriptor distributed by a service provider, the decryptor device capable of decrypting content from several different content provider devices; and

a controller for controlling the decryptor and for communicating with the service provider, thereby providing the subscriber access to the content data, wherein the content descriptor is distributed to the apparatus temporally independent of the distribution of the encrypted content.

21. The apparatus according to claim 20, further comprising a logging device for logging access to the content data, wherein the controller controls the logging of access to the content data by the logging device according to rules contained in the content descriptor.

22. The apparatus according to claim 20, further comprising a content descriptor table containing content descriptors from the service provider, wherein the controller determines whether a content descriptor in the content descriptor table corresponds to the received encrypted content.

23. The apparatus according to claim 20, further comprising a conversion device for converting the decrypted content data into a displayable form for display on a display device.

24. The apparatus according to claim 20, further comprising communication control logic, the communication control logic being responsive to the controller to communicate with the service provider.

25. The apparatus according to claim 20, further comprising:

a menu generator for generating a menu, having a plurality of operations, to be displayed on a display device; and

a menu navigator for providing a user selection of said operations to said controller.

26. The apparatus according to claim 20, further comprising a processor for processing the decrypted content data to inhibit distribution of the decrypted content to more than one display device.

* * * * *

IX. RELATED PROCEEDINGS APPENDIX

There are no related proceedings.